User's Manual

78K0/Fx2 - CAN it!

Demonstration Kit for the F_Line Family

Global Document No. U18414EE1V1UM00 (1st edition)
Document No. 78K0FX2CANITV101
Date Published June 2006

Global Document No. U18414EE1V1UM00 (1st edition)

CAUTION

This is a Test- and Measurement equipment with possibility to be significantly altered by user through hardware enhancements/modifications and/or test or application software. Thus, with respect to Council Directive 89/336/EEC (Directive on compliance with the EMC protection requirements), this equipment has no autonomous function. Consequently this equipment is not marked by the CE-symbol.

EEDT-ST-0005-10



Redemption of Waste Electrical and Electronic Equipment (WEEE) in accordance with legal regulations applicable in the European Union only: This equipment (including all accessories) is not intended for household use. After use the equipment cannot be disposed of as household waste. NEC Electronics (Europe) GmbH offers to take back the equipment. All you need to do is register at www.eu.necel.com/weee.

Global Document No. U18414EE1V1UM00 (1st edition)

Revision History

Date	Revision	Chapter	Description
16-12-2005	V1.00	---	First release
21-06-2006	V1.01	4.2	Correction of CAN connector CN8

Global Document No. U18414EE1V1UM00 (1st edition)

Table of Contents

Global Document No. U18414EE1V1UM00 (1st edition)

Global Document No. U18414EE1V1UM00 (1st edition)

List of Figures

List of Tables

1. Introduction

78K0/Fx2 – CAN it! is a demonstration kit for the NEC's F_Line 78K0/Fx2 microcontroller family. It supports On-Board debugging, FLASH programming and real time execution of application programs up to 4 kBytes of program code for the 78K0/FF2 microcontroller. The board is prepared to be connected to user hardware parts such as digital I/O or analogue signals.

1.1 Main features of *78K0/Fx2 – CAN it!*

- Easy to use device demonstration capabilities
  78K0/Fx2 – CAN it! contains elements to easily demonstrate simple I/O-functions, i.e. push buttons, 12*2 character LC display, LED output, AD reference voltage, I/O lines, UART serial interface, LIN and CAN serial interfaces.

- On-Board debug function
  The *78K0/Fx2 – CAN it!* supports an On-Board debug function by using the IAR C-SPY debugger. It allows FLASH downloading and standard debug functions i.e. code execution, single stepping, breakpoints, memory manipulation etc.

- Power supply via USB interface
  78K0/Fx2 – CAN it! is powered via USB interface, no separate power supply is needed.

- Character LCD module
  78K0/Fx2 – CAN it! provides a 12*2 character LC display, allowing the implementation of human / machine interfaces, comfortable input / output functions, output of measurement values, output of status information etc.

- FPL3, FLASH programming software
  A windows based FLASH programming software allows to select and download application programs to the *78K0/Fx2 – CAN it!* board for evaluation purposes.

- Analogue to digital signal conversion is supported

- Various input / output signals available, such as

  - All I/O ports prepared to be connected to user hardware
  - Timer input / output signals
  - Two or three wire serial I/O
  - UART interface, via USB UART chip FT232
  - High Speed CAN bus interface with transceiver PCA82C250
  - LIN bus support, via TJA 1020 transceiver
  - 16 analogue input lines
  - Temperature sensor KTY13-5
  - 4 I/O ports connected to LED
  - 2 push buttons prepared for external interrupt generation

- The IAR Embedded Workbench for 78K0/78K0S and the IAR C-SPY debugger / simulator are included. These packages are restricted in such that maximum program code size is limited to 4 kByte of program code.

- Full documentation is included for the NEC 78K0/FF2 microcontroller, IAR Systems Embedded Workbench, IAR Systems C-SPY debugger / simulator and the NEC FPL3 FLASH programming software.

***78K0/Fx2 – CAN it!* is not intended for code development. NEC does not allow and does not support in any way any attempt to use *78K0/Fx2 – CAN it!* in a commercial or technical product.**

1.2 System requirements

HOST PC	A PC supporting Windows 98SE, Windows ME, Windows 2000 or Windows XP is required for the IAR Systems Embedded Workbench demo-version and the FPL3 FLASH programming software. Pentium 166 MHz (at least), 128 MB of RAM, 256-color display (1024 * 768), mouse, CD-ROM drive and 200 Mbytes of free hard disk space are required to install the tool packages. Above listed requirements are valid for the IAR Systems Embedded Workbench and the FPL3 FLASH programming software.
Host interface	USB interface that enables communication based on USB (Ver1.1 or later)

1.3 Package contents

Please verify that you have received all parts listed in the package contents list attached to the *78K0/Fx2 – CAN it!* package. If any part is missing or seems to be damaged, please contact the dealer from whom you received your *78K0/Fx2 – CAN it!*.

Note: Updates of the IAR Embedded Workbench for 78K, FP3 FLASH programming software, documentation and/or utilities for *78K0/Fx2 – CAN it!*, if available, may be downloaded from the NEC WEB page(s) at **http://www.eu.necel.com/updates**

1.4 Trademarks

IAR Embedded Workbench, visualSTATE, IAR MakeApp and C-SPY are registered trademarks of IAR Systems AB. Microsoft and Windows are registered trademarks of Microsoft Corporation. Adobe and Acrobat Reader are registered trademarks of Adobe Systems Incorporated.

All other product names are trademarks or registered trademarks of their respective owners.

Global Document No. U18414EE1V1UM00 (1st edition)

2. *78K0/Fx2 – CAN it!* system configuration

The *78K0/Fx2 – CAN it!* system configuration is given in the diagram below:

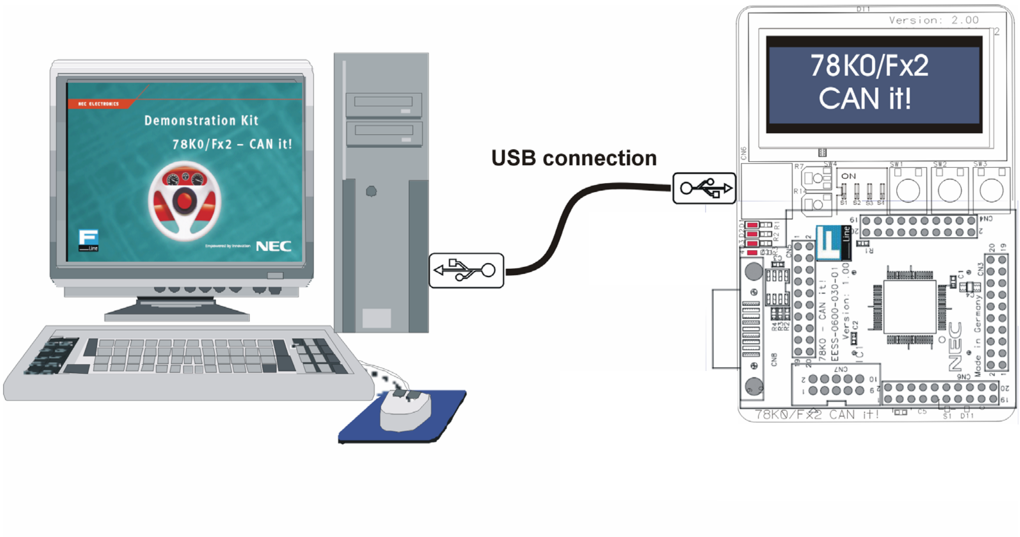


Figure 1: *78K0/Fx2 – CAN it!* system configuration

2.1 *78K0/Fx2 – CAN it!*

78K0/Fx2 – CAN it! is a demonstration kit for the 78K0/FF2 8-Bit microcontroller. The *78K0/Fx2 – CAN it!* board is connected to the host system via USB interface cable. The host system may be used for On-Board debugging of application software by using the IAR C-SPY debugger or the programming of the 78K0/FF2 internal FLASH memory by using the FPL3 programming GUI, to allow execution of application programs on the *78K0/Fx2 – CAN it!* starterkit hardware.

78K0/Fx2 – CAN it! runs the microcontroller at 12.0000 MHz operating speed. Sub-clock is provided with 32.768 kHz.

2.2 Host computer

The USB host interface enables communication to the *78K0/Fx2 – CAN it!* board. The USB UART chip FT232 allows application software to access the USB device in the same way as it would access a standard RS232 interface. The FTDI's Virtual COM Port (VCP) driver appears to the windows system as an extra Com Port, in addition to any existing hardware Com Ports.

2.3 Power supply via USB interface

78K0/Fx2 – CAN it! is powered by USB interface, no separate power supply is needed. The USB interface provides the *78K0/Fx2 – CAN it!* board with 5V supply voltage.

3. *78K0/Fx2 – CAN it!* baseboard components

The *78K0/Fx2 – CAN it!* baseboard is equipped with push buttons, a 12*2 character LC display, LED's and with several connectors in order to be connected to host computers, FLASH programmer, LIN and High Speed CAN busses.

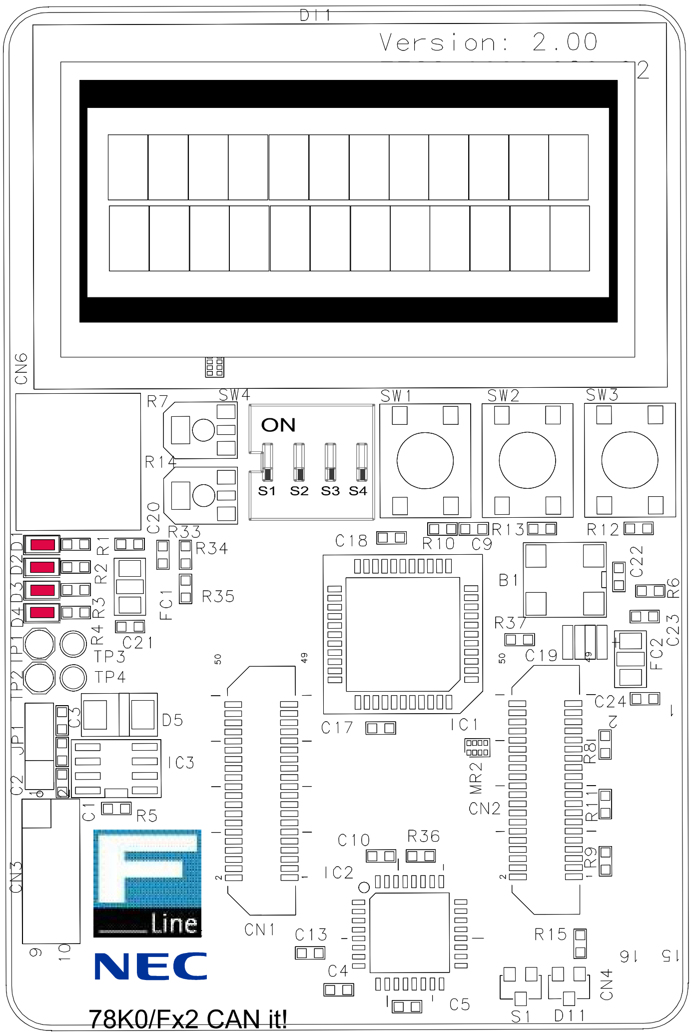


Figure 2: *78K0/Fx2 – CAN it!* baseboard connectors, switches and LED's

Some of the *78K0/Fx2 – CAN it!* components are free for user application hardware and software. Please read the user's manual of the 78K0/FF2 device carefully to get information about the electrical specification of the available I/O ports before you connect any external signal to the *78K0/Fx2 – CAN it!* board.

3.1 RESET button SW1

SW1 is a reset button. It activates the power on reset. It is connected to the reset input of the 78K0/FF2 CPU module.

3.2 User button SW2

SW2 is a push button connecting V_{cc} to external interrupt input INTP0 of the microcontroller. This is equal to port P120 of the 78K0/FF2 device. The port may be programmed to generate interrupt INTP0. The necessary initialisation for this purpose is described in the user's manual of the 78K0/FF2 device. The port is connected to a 4.7K pull down resistor.

3.3 User button SW3

SW3 is a push button connecting V_{cc} to external interrupt input INTP1 of the microcontroller. This is equal to port P30 of the 78K0/FF2 device. The port may be programmed to generate interrupt INTP1. The necessary initialisation for this purpose is described in the user's manual of the 78K0/FF2 device. The port is connected to a 4.7K pull down resistor.

3.4 Configuration switch SW4

The different operation modes of the *78K0/Fx2 – CAN it!* board can be set by SW4 switches S1-S4.

SW4	Factory settings	Operation Mode
S1	OFF	Normal operation mode
S2	OFF	OCD disabled
S3	OFF	UART60 select
S4	OFF	LIN bus disabled

Table 1: Configuration switch SW4, factory settings

3.4.1 Operation mode selection SW4/S1

SW4 switch S1 controls the operation mode of the *78K0/Fx2 – CAN it!* board. Setting SW4/S1 to ON allows to reprogram the internal FLASH memory of the 78K0/FF2 device using the FPL3 FLASH programming software.

SW4, S1	Operation mode
OFF (default)	Normal operation mode
ON	FLASH memory programming mode

Table 2: Operation mode selection SW4/S1

Within normal operation mode the user program stored in the FLASH memory of 78K0/FF2 device is executed.

3.4.2 On-Chip debug mode selection SW4/S2

SW4 switch S2 controls the On-Chip debug function of the 78K0/FF2 device. Setting switch S2 to ON allows to use the On-Board debug function of the *78K0/Fx2 – CAN it!* board or alternative connecting the QB-78K0MINI On-Chip debug emulator to the 78K0/FF2 CPU module, connector CN7.

SW4, S2	Operation Mode
OFF (default)	OCD disabled
ON	OCD enabled

Table 3: OCD mode selection SW4/S2

For more details on how to configure *78K0/Fx2 – CAN it!* in order to use On-Chip debugging please refer to **CHAPTER 5, ON-CHIP DEBUGGING**.

3.4.3 UART selection SW4/S3

SW4 switch S3 specifies the corresponding UART signals of the 78K0/FF2 device connected to the FT232 interface lines.

SW4, S3	78K0/FF2
OFF (default)	UART60 select
ON	UART61 select

Table 4: UART selection SW4/S3

3.4.4 UART/ LIN mode SW4/S4

SW4 switch S4 controls the serial communication mode of *78K0/Fx2 – CAN it!* board. The corresponding UART signals RxD and TxD selected by SW4/S3 are connected to the TJA1020 LIN transceiver when setting SW4/S4 to ON. Setting SW4/S4 to OFF connects the UART signals to the FT232 interface lines.

SW4, S4	Operation mode
OFF (default)	UART mode
ON	LIN mode

Table 5: UART / LIN selection SW4/S4

3.5 LIN plug JP1

JP1 is a 3 pin connector for the LIN bus, connected to the transceiver TJA1020.

JP1	Signal	Description
1	BAT	Reference voltage for the LIN bus level
2	LIN	LIN bus line
3	GND	Ground

Table 6: LIN plug JP1

3.6 USB interface connector CN6

The CN6 connector allows connecting the IAR C-SPY debugger or FPL3 FLASH programming software to the *78K0/Fx2 – CAN it!* board in order to debug application software or program the 78K0/FF2 internal FLASH memory. The board power supply of 5V is also provided by connector CN6.

For standard communication to a host system, i.e. by using a terminal program, the input/output signals of UART60 respectively UART61 of the 78K0/FF2 device are connected to CN6. Please configure switch SW4 accordingly to use this mode.

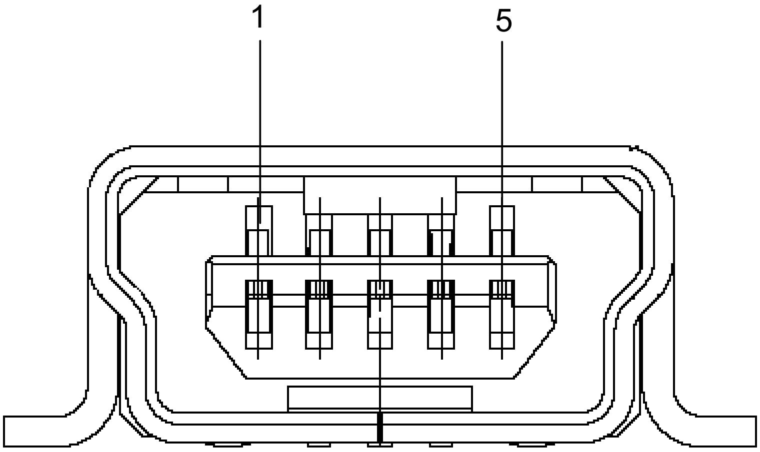


Figure 3: Connector CN6, USB Mini-B Type Host Connector Pin Configuration

USB Connector CN6	Signal Name
1	VBUS
2	DM
3	DP
4	N.C.
5	GNDBUS

Table 7: Pin Configuration of USB Connector CN6

For connection with the host machine, use a USB cable (Mini-B type). For confirmation, NEC Electronics used only the USB cable delivered with the *78K0/Fx2 – CAN it!* board.

3.7 Connector CN4

CN4 connector (not assembled) allows connecting the PG-FP4 flash programmer to *78K0/Fx2 – CAN it!* board in order to program application programs into the microcontroller internal FLASH memory. Please note, PG-FP4 is a separate product from NEC and it is not included in this package.

CN4	Signal
1	GND
2	RESET
3	SI
4	V_{CC}
5	SO
6	N.C.
7	SCK
8	N.C.
9	N.C.
10	N.C.
11	N.C.
12	FLMD1
13	N.C.
14	FLMD0
15	N.C.
16	N.C.

Table 8: CN4, PG-FP4 connector

When using PG-FP4, the programming interface to the 78K0/FF2 device must be set to clock serial interface, CSI. Set configuration of switch SW4 of the *78K0/Fx2 – CAN it!* board to the following:

SW4	Setting
S1	OFF
S2	OFF
S3	OFF
S4	OFF

Table 9: SW4 configuration when using PG-FP4

3.8 External Potentiometer R7

A 47K potentiometer R7 is connected between Vcc and ground. The potentiometer arm is connected to the V0 signal of the LCD module. It controls the operating voltage - contrast adjustment - of the display.

3.9 Display D1, 12*2 character LC display

The *78K0/Fx2 – CAN it!* board is equipped with a character LC display. The display font is equal to 12 character words at 2 lines. The LCD module contains about a character generator ROM - including predefined standard characters - and a character RAM where the user can define its own characters. The display is connected to the 78K0/FF2 device via three control lines and eight data lines.

Display Pin	Display Signal	Port 78K0/FF2	*78K0/Fx2- CAN it!* baseboard
1	Vss	-	GND
2	Vcc	-	Vcc
3	V0	-	R7 potentiometer arm
4	RS	P66	-
5	R/W	P65	-
6	E	P64	-
7	DB0	P40	-
8	DB1	P41	-
9	DB2	P42	-
10	DB3	P43	-
11	DB4	P44	-
12	DB5	P45	-
13	DB6	P46	-
14	DB7	P47	-
15	A	-	Vcc via diode

Table 10: Display connections

For more details about the LC display specification, commands and character table, please refer to the corresponding User's Manual "WH1202A-NFA-ET.pdf" located in the `/doc` folder of the *78K0/Fx2 – CAN it!*.

3.10 External Potentiometer R14

A 47K potentiometer R14 is connected between Vcc and ground. The potentiometer arm is connected to port P80 of 78K0/FF2 device. This is equal to the ANI0 analogue input.

3.11 AD converter reference voltage input

A 1.2V reference voltage is supplied to the ANI1 analogue input, equal to port P81 of 78K0/FF2 device.

3.12 Temperature sensor

For temperature measurement and primarily as an application example a silicon temperature sensor KTY13-5 is connected to the ANI2 analogue input, equal to port P82 of 78K0/FF2 device.

The temperature sensor has a resistor range of $R_{25\,min}$ = 1950 • and $R_{25\,max}$ = 1990 • at 25 degrees centigrade, with I_{OP} = 1mA. The distribution of the temperature factor k_T is shown in the table below:

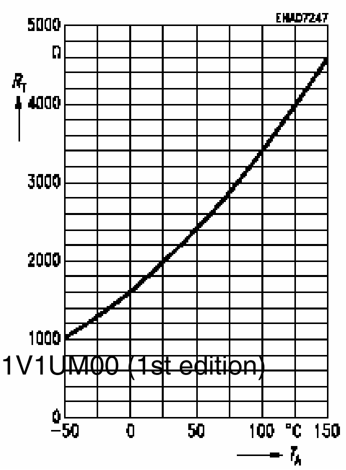
T_A	k_T		
°C	min.	typ.	max.
– 50	0.506	0.518	0.530
– 40	0.559	0.570	0.581
– 30	0.615	0.625	0.635
– 20	0.676	0.685	0.694
– 10	0.741	0.748	0.755
0	0.810	0.815	0.821
10	0.883	0.886	0.890
20	0.960	0.961	0.962
25	1.0 [1]		
30	1.039	1.040	1.041
40	1.119	1.123	1.126
50	1.204	1.209	1.215
60	1.291	1.300	1.308
70	1.383	1.394	1.405
80	1.478	1.492	1.506
90	1.577	1.594	1.611
100	1.680	1.700	1.720
110	1.786	1.810	1.833
120	1.896	1.923	1.951
130	2.010	2.041	2.072
140	2.093	2.128	2.163
150	2.196	2.235	2.274

1) Normalising point

Table 11: Distribution of temperature factor k_T

The sensor resistance can be calculated as following: $R_T = k_T * R_{25} = •(T_A)$

I_B = 1 mA; Example: R_{25} = 2000 Ω



The temperature at the sensor can be calculated from the change in the sensors resistance from the following equation, which approximates the characteristic curve:

$$T = \left(25 + \frac{\sqrt{\alpha^2 - 4 \times \beta + 4 \times \beta \times k_T} - \alpha}{2 \times \beta} \right) °C$$

with: $\alpha = 7{,}88 \times 10^{-3} \times K^{-1}$

$\beta = 1{,}937 \times 10^{-5} \times K^{-2}$

$k_T = \frac{R_T}{R_{25}}$

3.13 External LED's D1–D4

LED's D1 to D4 are connected to port P5 of the 78K0/FF2 device. A low signal output at each port switches the corresponding LED on.

LED	Port 78K0/FF2
D1	P50
D2	P51
D3	P52
D4	P53

Table 12: LED D1–D4 connection

4. *78K0/Fx2 – CAN it!* CPU module components

The 78K0/FF2 CPU module is equipped with 4 connectors in order to be connected to user defined hardware. Additionally the 78K0/FF2 CPU module contains about connector CN7 for On-Chip debugging purpose and connector CN8 for CAN communication.

4.1 *78K0/Fx2 – CAN it!* CPU module

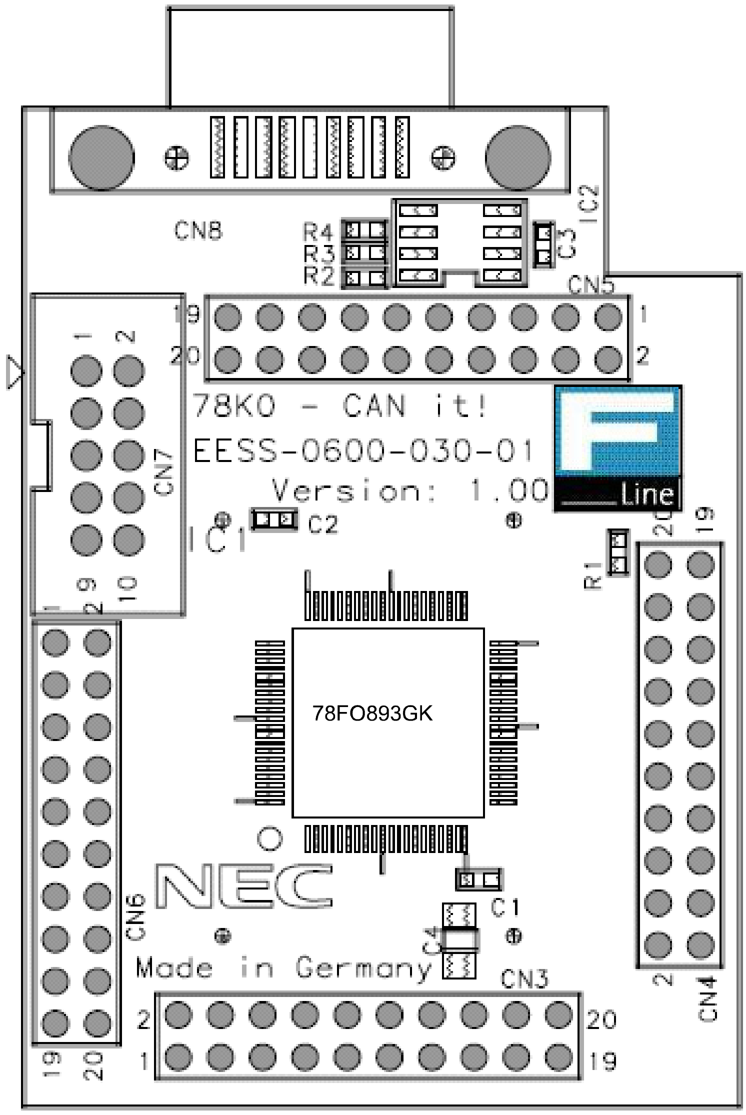


Figure 4: *78K0/Fx2 – CAN it!* CPU module components

4.1.1 External connector CN3, CN4, CN5, and CN6

CN3, CN4, CN5, and CN6 are connectors for external user hardware.

CN3	Port 78K0/FF2
1	P120, INTP0, EXLVI
2	P47
3	P46
4	P45
5	P44
6	P43
7	P42
8	P41
9	P40
10	/RESET
11	P124, XT2, EXCLKS
12	P123, XT1
13	FLDM0
14	P122, X2, EXCLK
15	P121, X1
16	REGC
17	Vss
18	EVss
19	VDD
20	EVDD

CN4	Port 78K0/FF2
1	P60
2	P61
3	P62
4	P63
5	P33, TI51, TO51, INTP4
6	P64
7	P65
8	P66
9	P67
10	P130
11	P76, SCK11
12	P75, SI11
13	P74, SO11
14	P73, BUZ, INTP7
15	P72, PCL, INTP6
16	P71, CRXD
17	P70, CTXD
18	P06, TI011, TO01
19	P05, SSI11, TI001
20	P32, TI012, TO02, INTP3

CN5	Port 78K0/FF2
1	P31, TI002, INTP2
2	P50
3	P51
4	P52
5	P53
6	P30, INTP1
7	P17, TI50, TO50
8	P16, TOH1, INTP5
9	P15, TOH0
10	P14, RXD60
11	P13, TXD60
12	P12, SO10
13	P11, SI10, RXD61
14	P10, SCK10, TXD61
15	P54
16	P55
17	P56
18	P57
19	AVREF
20	AVSS

CN6	Port 78K0/FF2
1	P97, ANI15
2	P96, ANI14
3	P95, ANI13
4	P94, ANI12
5	P93, ANI11
6	P92, ANI10
7	P91, ANI9
8	P90, ANI8
9	P87, ANI7
10	P86, ANI6
11	P85, ANI5
12	P84, ANI4
13	P83, ANI3
14	P82, ANI2
15	P81, ANI1
16	P80, ANI0
17	P01, TI010, TO00
18	P00, TI000
19	P132, TI013, TO03
20	P131, TI003

Table 13: CN3, CN4, CN5, and CN6 connection to 78K0/FF2

4.2 High Speed CAN connector CN8

CN8 is a D-SUB 9 connector for High Speed CAN with CiA standard pin assignment. The used transceiver is the PCA82C250 with bus termination. The transceiver is connected to the CAN interface of the 78K0/FF2 device, whereby the standby mode control is selected by port P67.

CN8	Function
1	N.C.
2	CANL
3	GND
4	N.C.
5	N.C.
6	N.C.
7	CANH
8	N.C.
9	N.C.

Table 14: CAN connector CN8

4.2.1 OCD connector CN7

Connector CN7 (not assembled) allows connecting the QB-78K0MINI On-Chip debug emulator to the *78K0/Fx2 – CAN it!* board in order to use the On-Chip debug function of the 78K0/FF2 device. QB-78K0MINI is a separate product from NEC and it is not included in this starterkit package.

CN7	Function
1	RESET_IN
2	RESET_OUT
3	FLMD0
4	VDD_IN
5	X2
6	GND
7	X1
8	GND
9	N.C.
10	N.C.

Table 15: OCD connector CN7

To enable On-Chip debugging by using the QB-78K0MINI, please set switch SW4 to the following configuration:

SW4	Setting
S1	OFF
S2	ON
S3	OFF
S4	OFF

Table 16: SW4 configuration for OCD via QB-78K0MINI

For more details on how to configure *78K0/Fx2 – CAN it!* in order to use On-Chip debugging please refer to **CHAPTER 5, ON-CHIP DEBUGGING**.

5. On-Chip debugging

The *78K0/Fx2 – CAN it!* board offers two possibilities to use On-Chip debugging (OCD). The On-Board debug function of *78K0/Fx2 – CAN it!* allows On-Chip debugging without a need of external debug hardware. Within this mode the default USB / UART connection to the Host computer is used as debug interface. All standard debug functions are available in the On-Board debugging mode like FLASH programming / downloading, code execution, single stepping, breakpoints, memory manipulation etc.

Additionally *78K0/Fx2 – CAN it!* supports the QB-78K0MINI On-Chip debug emulator in order to use On-Chip debug function of the 78K0/FF2 device. The system configuration for On-Chip debugging is shown in figure below.

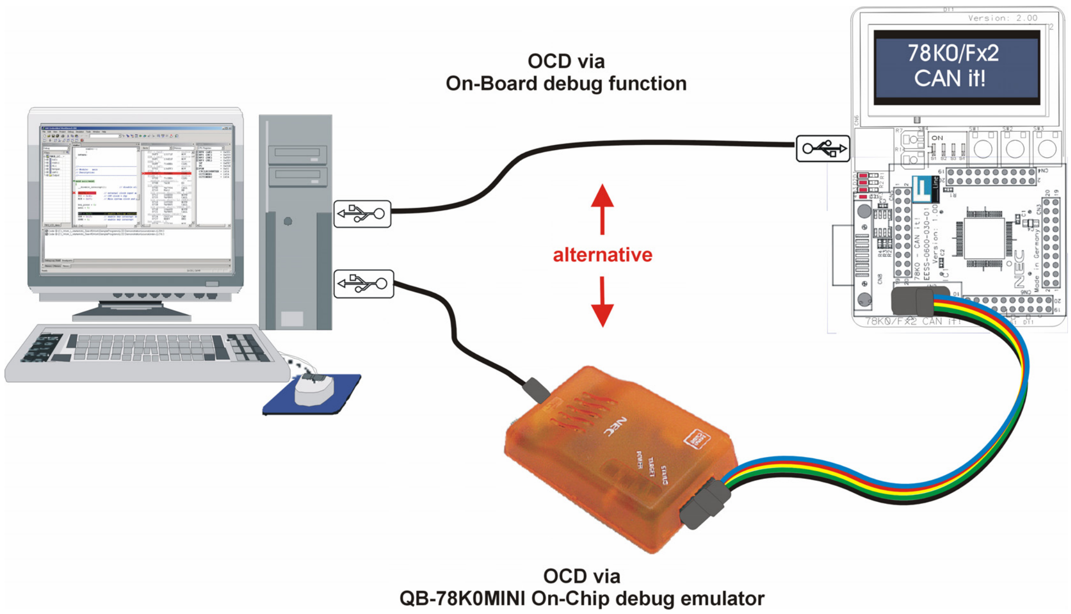


Figure 5: *78K0/Fx2 – CAN it!* configuration for On-Chip debugging

5.1 OCD via On-Board debug function

To operate the *78K0/Fx2 – CAN it!* board within the On-Board debug mode configure switch SW4 as following:

SW4	Setting
S1	ON
S2	ON
S3	OFF
S4	OFF

Table 17: SW4 configuration for OCD via On-Board debug function

5.2 OCD via QB-78K0MINI On-Chip debug emulator

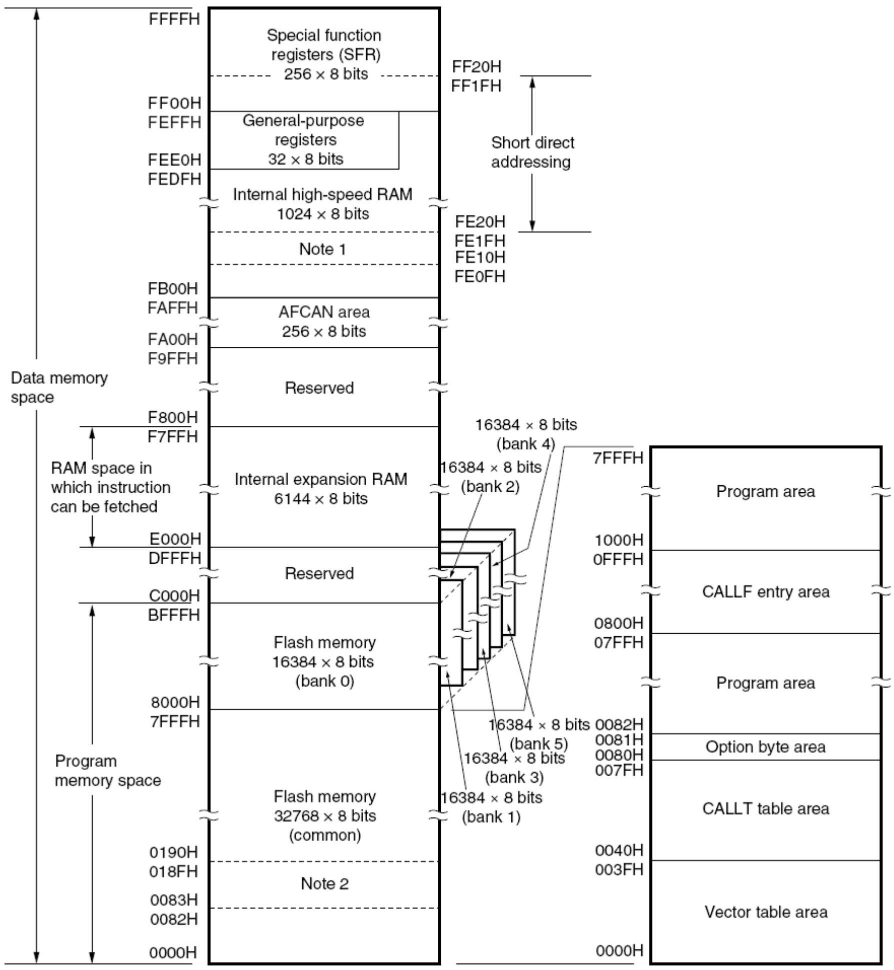
To operate the *78K0/Fx2 – CAN it!* board together with the QB-78K0MINI On-Chip debug emulator, configure switch SW4 as following:

SW4	Setting
S1	OFF
S2	ON
S3	OFF
S4	OFF

Table 18: SW4 configuration for OCD via QB-78K0MINI

5.3 78K0/FF2 memory map

The 78K0/FF2 memory layout is shown in the table below.



Notes 1. During on-chip debugging, use of this area is disabled since it is used as the user data backup area for communication.

2. During on-chip debugging, use of this area is disabled since it is used as the communication command area (256 bytes to 1 KB).

Table 19: 78K0/FF2 memory map

The *78K0/Fx2 – CAN it!* does not reserve any resources of the 78K0/FF2 device, consequently all available memory of the device is free for application software.

6. *78K0/Fx2 – CAN it!* installation and operation

6.1 Getting started

The IAR C-SPY debugger allows to download and debug application software on the *78K0/Fx2 – CAN it!* starterkit hardware. Additionally the FPL3 FLASH programming software can be used for simple FLASH programming of the 78K0/FF2 internal FLASH memory. As communication interface between the host computer and the *78K0/Fx2 – CAN it!* board a USB interface line is needed. Before you can download, debug or execute an application program, hardware and software must be installed properly.

6.1.1 CD-ROM contents

The CD-ROM shows following directory structure:

78K0/Fx2 – CAN it! (F:)	CD-ROM ROOT
Acrobat	- Acrobat Reader for 32Bit Windows OS
Device Files	- Device Files for *78K0/Fx2 – CAN it!*
DDF	… Device Description Files and … SFR Description Files for IAR C-SPY
NEC	… Device Files for IAR C-SPY
Doc	- Documentation
FPL3	- FPL3 FLASH programming software
Drivers	… USB driver
FPL3	… FPL3 setup directory
PRM	… PRM parameter files
IAR Embedded Workbench 78K0	- IAR Embedded Workbench for 78K0/78K0S
SamplePrograms	- Sample programs for *78K0/Fx2 – CAN it!*

Table 20: *78K0/Fx2 – CAN it!* CD-ROM directory structure

7. Hardware installation

After unpacking *78K0/Fx2 – CAN it!,* connect the board to your host computer using the provided USB interface cable. When *78K0/Fx2 – CAN it!* is connected, the USB driver needs to be installed on the host machine. Please refer to the following **CHAPTER 8 SOFTWARE INSTALLATION**.

8. Software installation

The *78K0/Fx2 – CAN it!* package comes with several software demo packages:

- IAR Systems Embedded Workbench for 78K0/78K0S, including C compiler, assembler, linker, librarian and IAR C-SPY debugger
- Device Files for *78K0/Fx2 – CAN it!*
- FPL FLASH programming software
- Sample programs

The IAR Systems Embedded Workbenches and the FPL3 FLASH programming GUI must be installed on your PC. For detailed installation hints, refer to the following chapters and to the corresponding documentation of the IAR Embedded Workbench.

8.1 IAR Systems Embedded Workbench for 78K0/78K0S installation

To install the IAR Systems Embedded Workbench for 78K0/K0S including C-SPY debugger, select the `SETUP` program in the directory `\IAR Embedded Workbench 78K\ew78k\` of the CDROM. The setup dialogues will guide you through the installation process.

8.2 Device file installation

To install the device files for the *78K0/Fx2 – CAN it!* in order to use the IAR C-SPY debugger, select the `SETUP` program in the directory `\Device Files\` of the CDROM. The setup dialogues will guide you through the installation process.

8.3 FPL3 FLASH programming GUI installation

To install the FPL3 FLASH programming GUI select the `SETUP` program in the directory `\FPL3\` of the CDROM. The setup dialogues will guide you through the installation process.

8.4 Sample program installation

To install the sample / demonstration programs for the *78K0/Fx2 – CAN it!* board select the `SETUP` program in the directory `\SamplePrograms\` of the CDROM. The setup dialogues will guide you through the installation process.

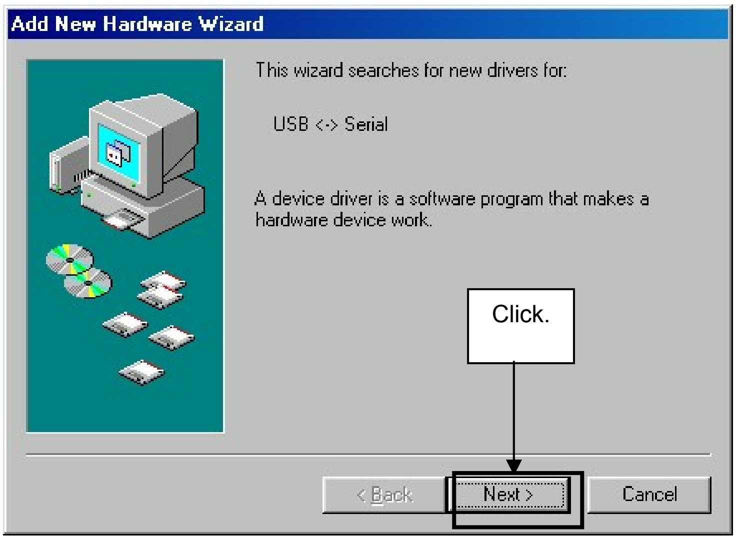
8.5 USB Driver Installation

In order to use the *78K0/Fx2 – CAN it!* board for On-Chip debugging or FLASH programming, the USB driver needs to be installed on the host machine. Install the driver according to the following procedure:

8.5.1 Installation on Windows 98SE/Me

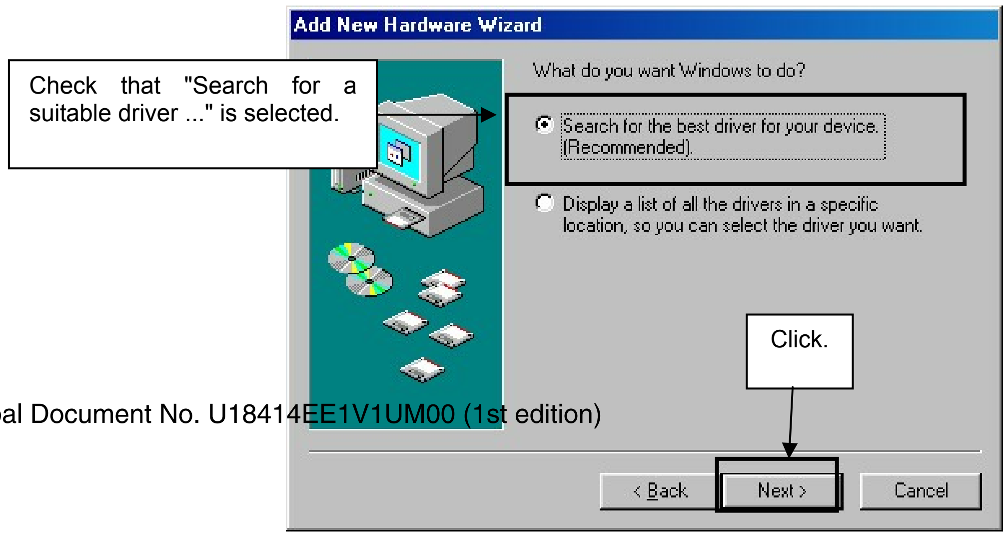
1. When the *78K0/Fx2 – CAN it!* board is connected with the host machine, the board is recognized by Plug and Play, and the wizard for adding new hardware is started. Click Next>.

Figure 6: Add New Hardware Wizard (Windows 98SE)



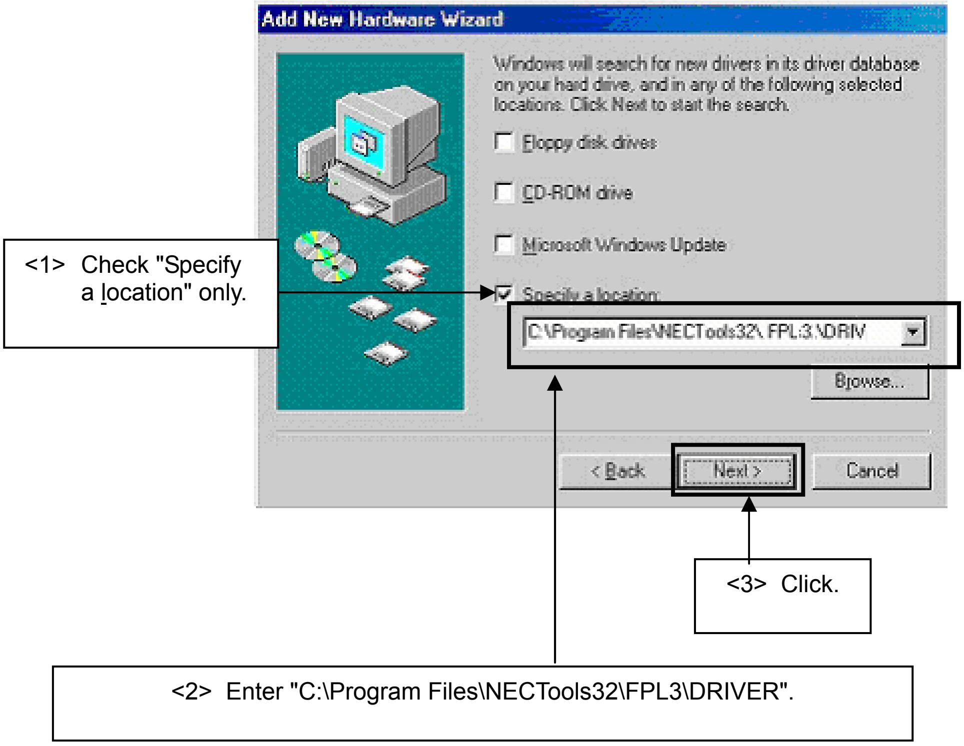
2. The window below is displayed. So, check that "Search for a suitable driver ..." is selected, then click Next>.

Figure 7: Search Method (Windows 98SE)



3. Check the "Specify a location" check box only and enter "C:\Program Files\NECTools32\FPL3\DRIVER" in the address bar, then click Next>.

Figure 8: Search Location Specification (Windows 98SE)



Remark If the installation destination folder is changed at the time of GUI software installation, enter "new-folder\DRIVER".

4. The window below is displayed. Click Next>.

Figure 9: Checking Driver to Be Installed (Windows 98SE)



5. When the window below is displayed, the installation of the USB driver is completed. Click Finish. The installation of the USB Serial Port driver is then automatically performed.

Figure 10: Installation Completion (Windows 98SE)



8.5.2 Installation on Windows 2000

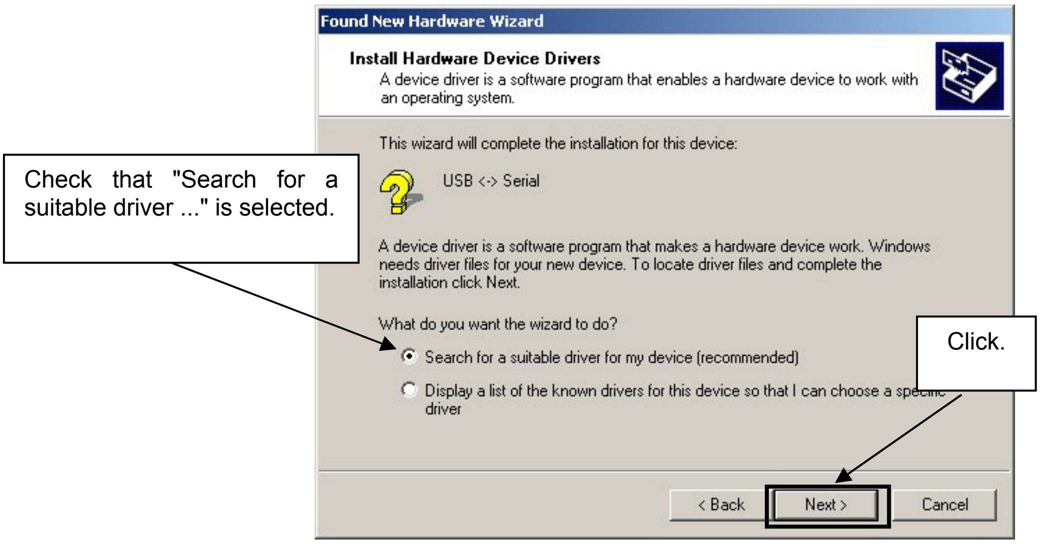
1. When the *78K0/Fx2 – CAN it!* board is connected with the host machine, the board is recognized by Plug and Play, and the wizard for finding new hardware is started. Click Next>.

Figure 11: Found New Hardware Wizard 1 (Windows 2000)



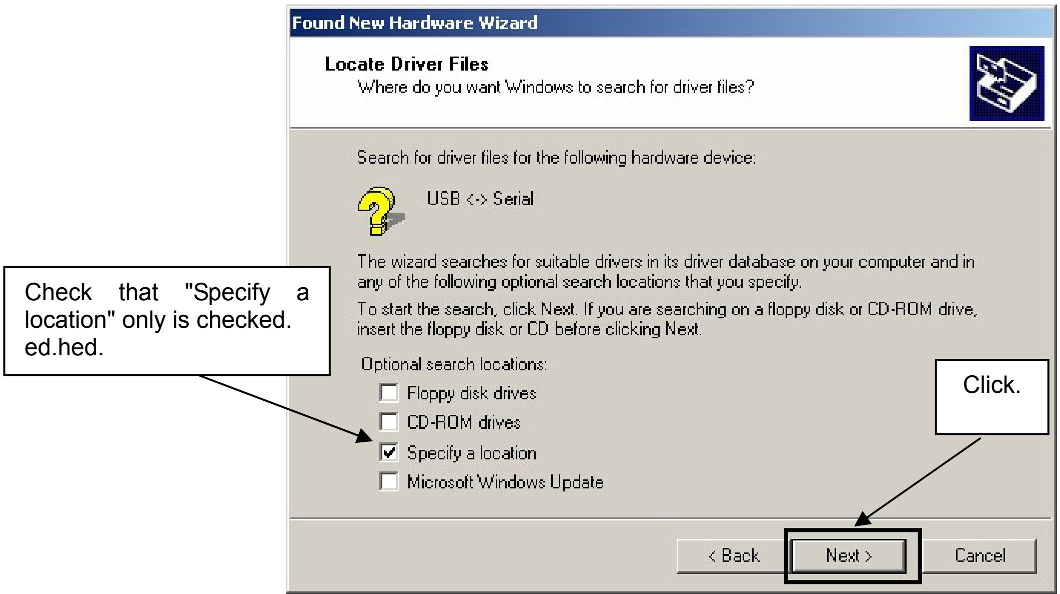
2. The window below is displayed. So, check that "Search for a suitable driver ..." is selected, then click Next>.

Figure 12: Search Method 1 (Windows 2000)



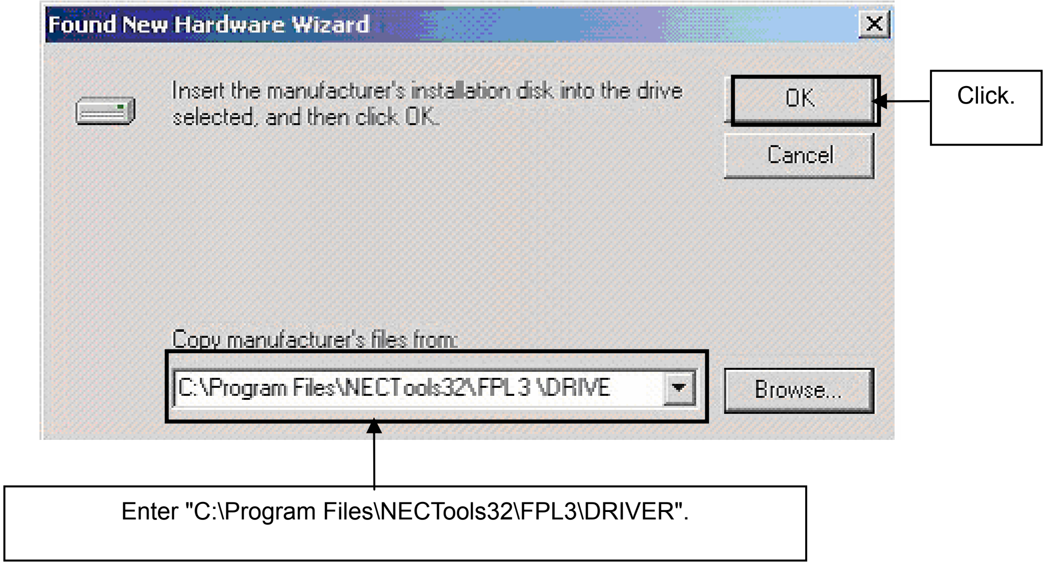
3. Check the "Specify a location" check box only, then click Next>.

Figure 13: Driver File Location 1 (Windows 2000)



4. Enter "C:\Program Files\NECTools32\FPL3\DRIVER" in the address bar, then click OK.

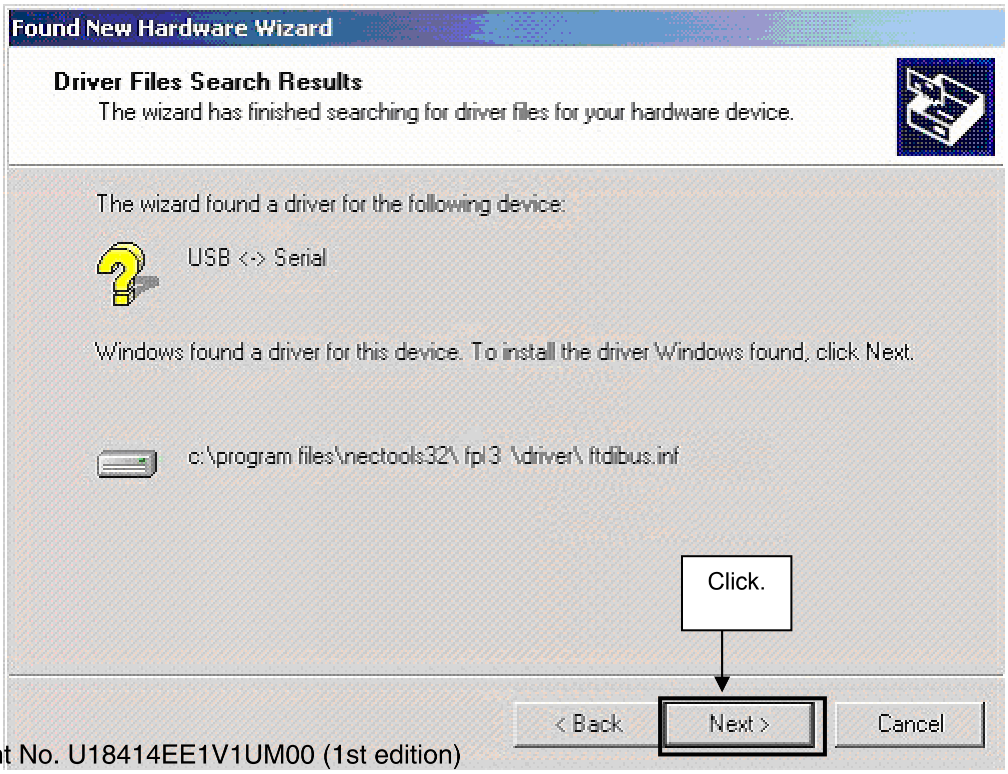
Figure 14: Address Specification 1 (Windows 2000)



Remark If the installation destination folder is changed at the time of GUI software installation, enter "new-folder\FPL3\DRIVER".

5. Click Next>.

Figure 15: Driver File Search 1 (Windows 2000)



6. Click Finish to complete the installation of the USB driver.

Figure 16: USB Driver Installation Completion 1 (Windows 2000)



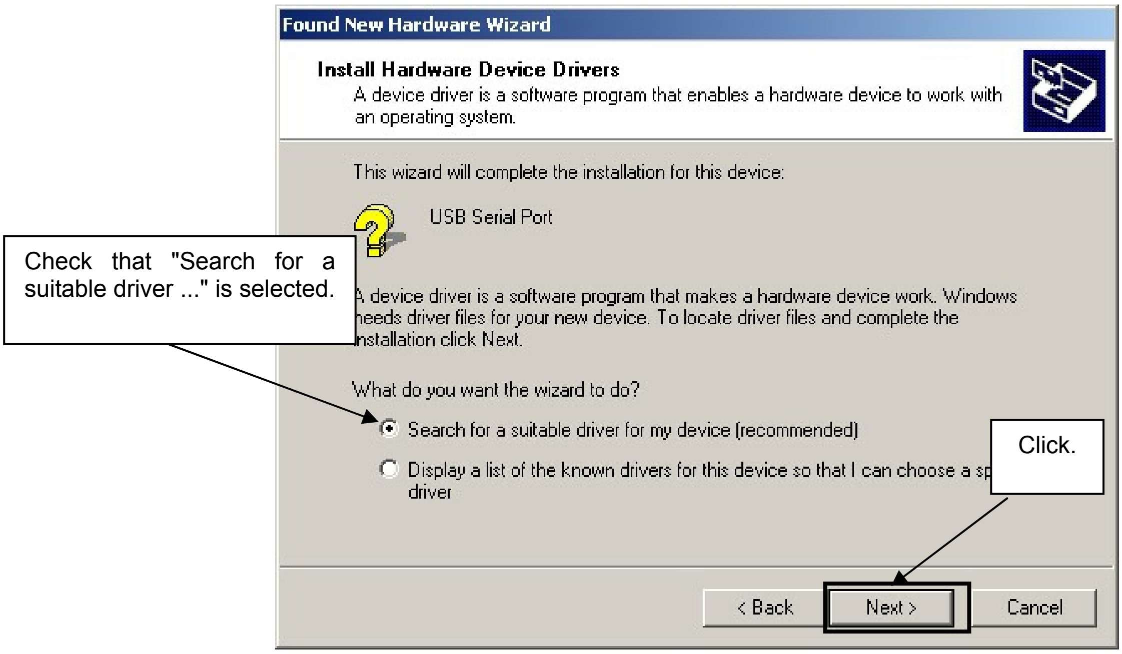
7. Proceed to the installation of the USB Serial Port driver. Click Next>.

Figure 17: Found New Hardware Wizard 2 (Windows 2000)



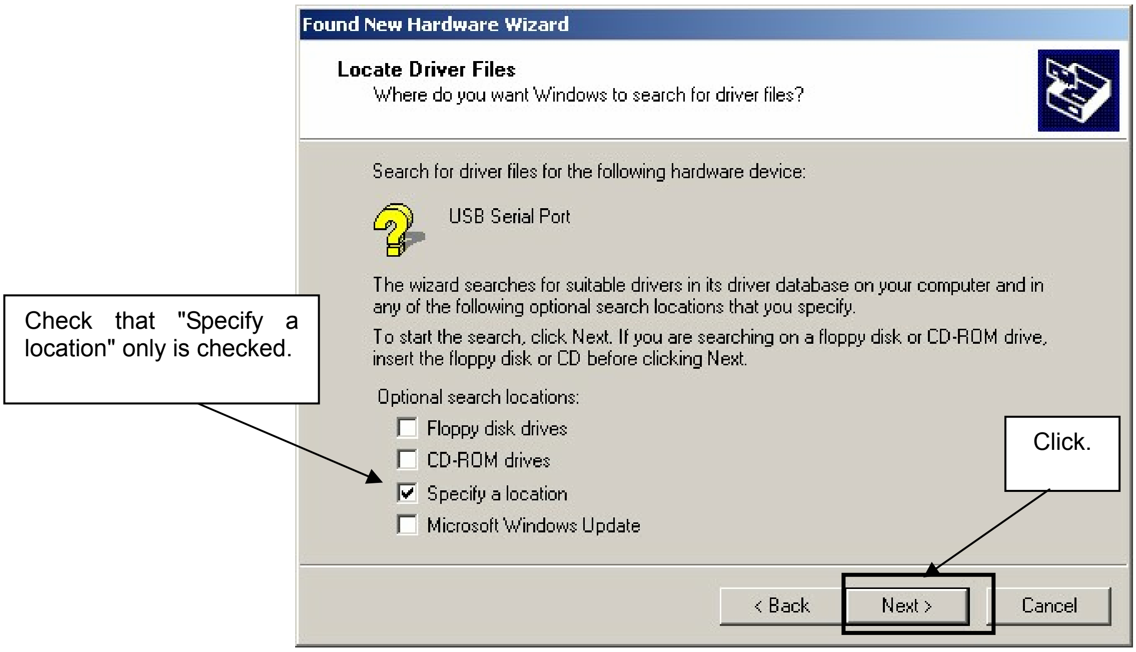
8. The window below is displayed. So, check that "Search for a suitable driver ..." is selected, then click Next>.

Figure 18: Search Method 2 (Windows 2000)



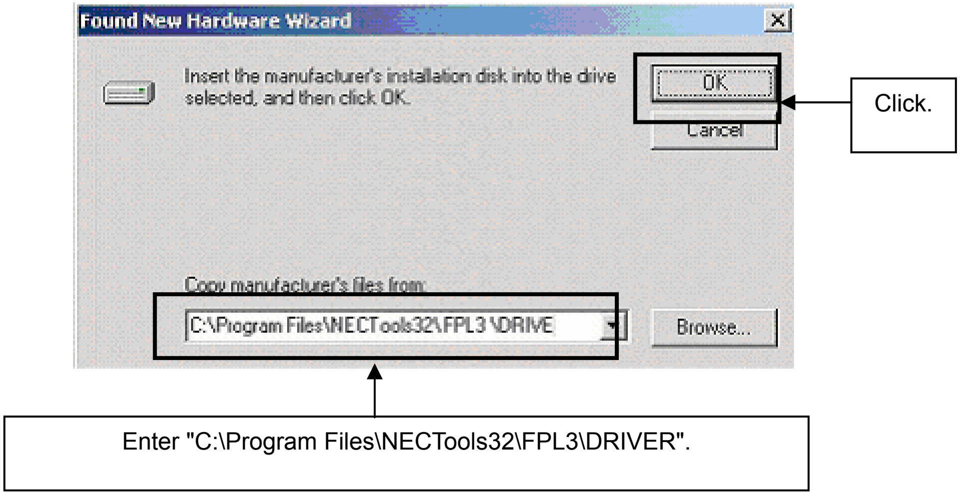
9. Check the "Specify a location" check box only, then click Next>.

Figure 19: Driver File Location 2 (Windows 2000)



10. Enter "C:\Program Files\NECTools32\FPL3\DRIVER" in the address bar, then click OK.

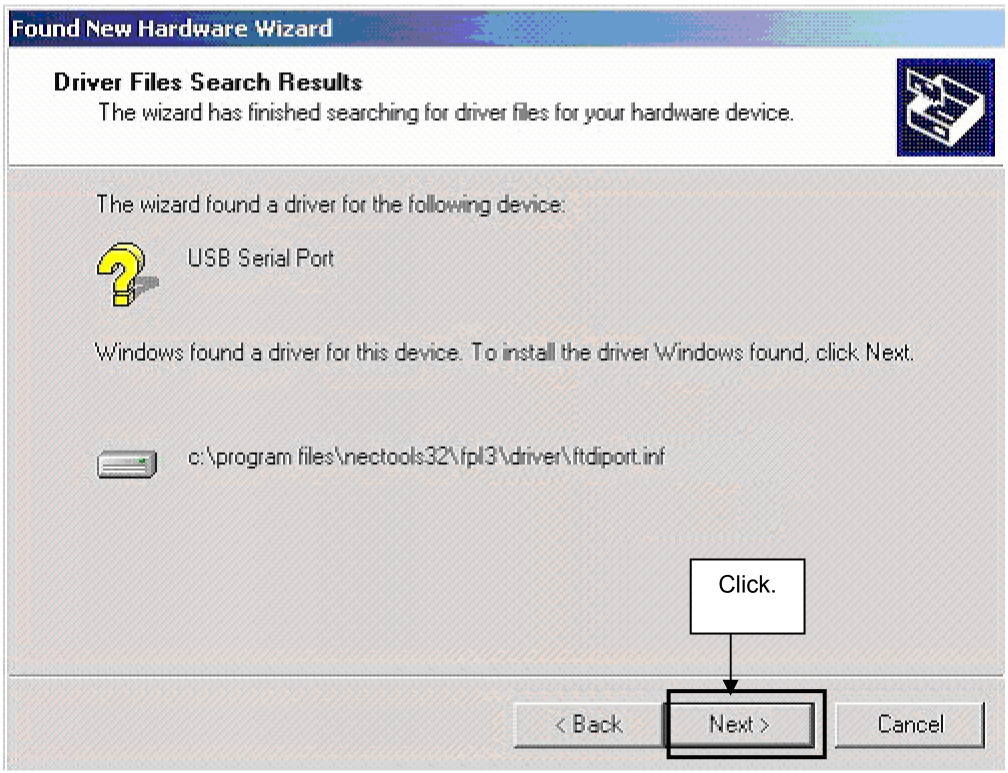
Figure 20: Address Specification 2 (Windows 2000)



Remark If the installation destination folder is changed at the time of GUI software installation, enter "new-folder\DRIVER".

11. Click Next>.

Figure 21: Driver File Search 2 (Windows 2000)



12. Click Finish to complete the installation of the USB driver.

Figure 22: USB Driver Installation Completion 2 (Windows 2000)



8.5.3 Installation on Windows XP

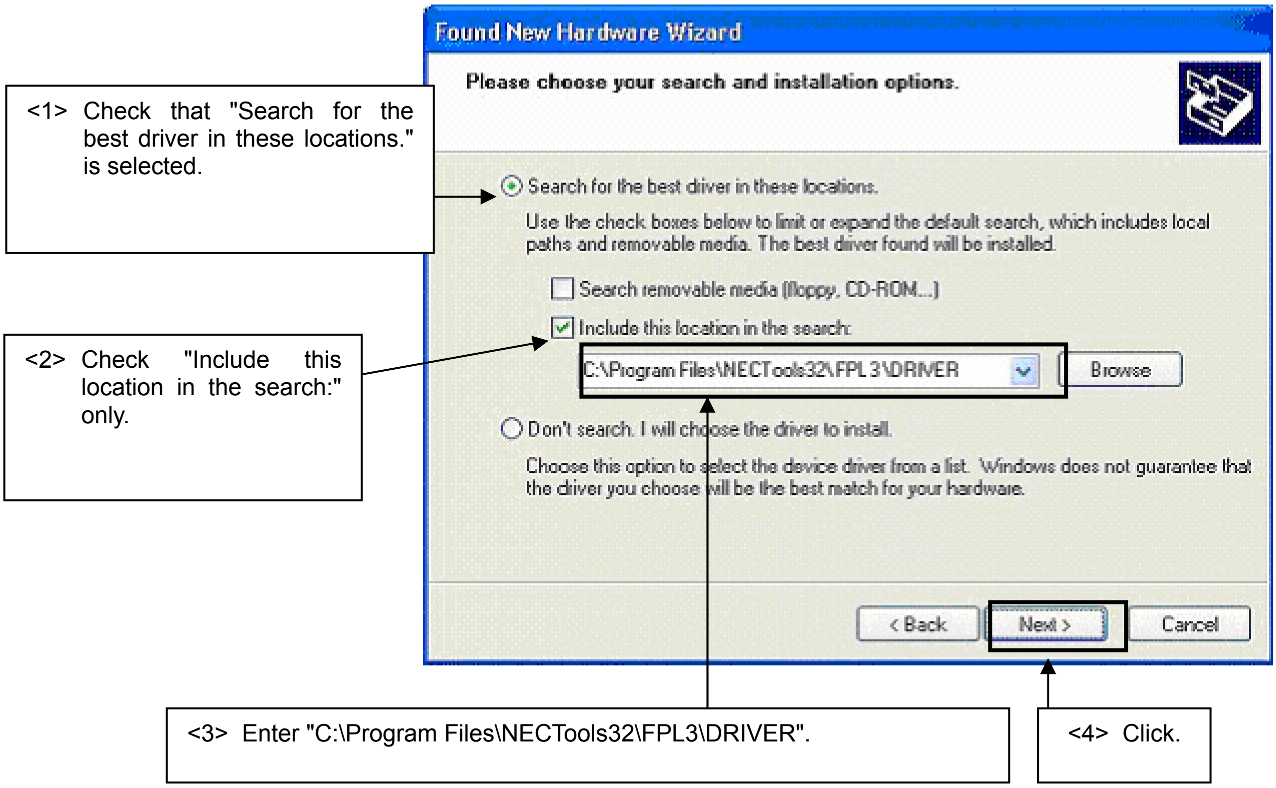
1. When the *78K0/Fx2 – CAN it!* board is connected with the host machine, the board is recognized by Plug and Play, and the wizard for finding new hardware is started. Check that "Install from a list or specific ..." is selected, then click Next>.

Figure 23: Found New Hardware Wizard 1 (Windows XP)



2. Check that "Search for the best driver in these locations." is selected. Check the "Include this location in the search:" check box and enter "C:\Program Files\NECTools32\FPL3\DRIVER" in the address bar, then click Next>.

Figure 24: Search Location Specification 3 (Windows XP)



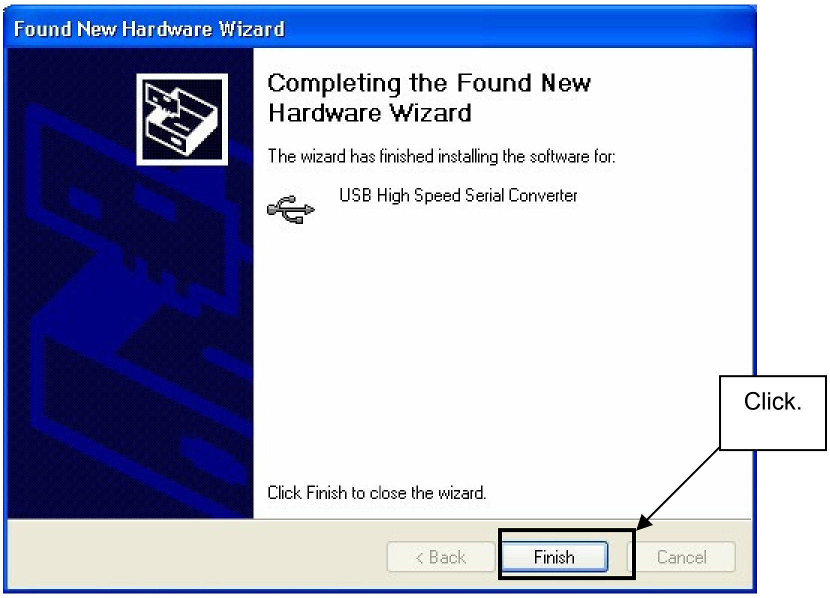
3. As shown below, "has not passed Windows Logo testing to verify its compatibility with Windows XP." is displayed. Click Continue Anyway.

Figure 25: Windows XP Logo Testing 3 (Windows XP)



4. When the window below is displayed, the installation of the USB driver is completed. Click Finish.

Figure 26: USB Driver Installation Completion 1 (Windows XP)



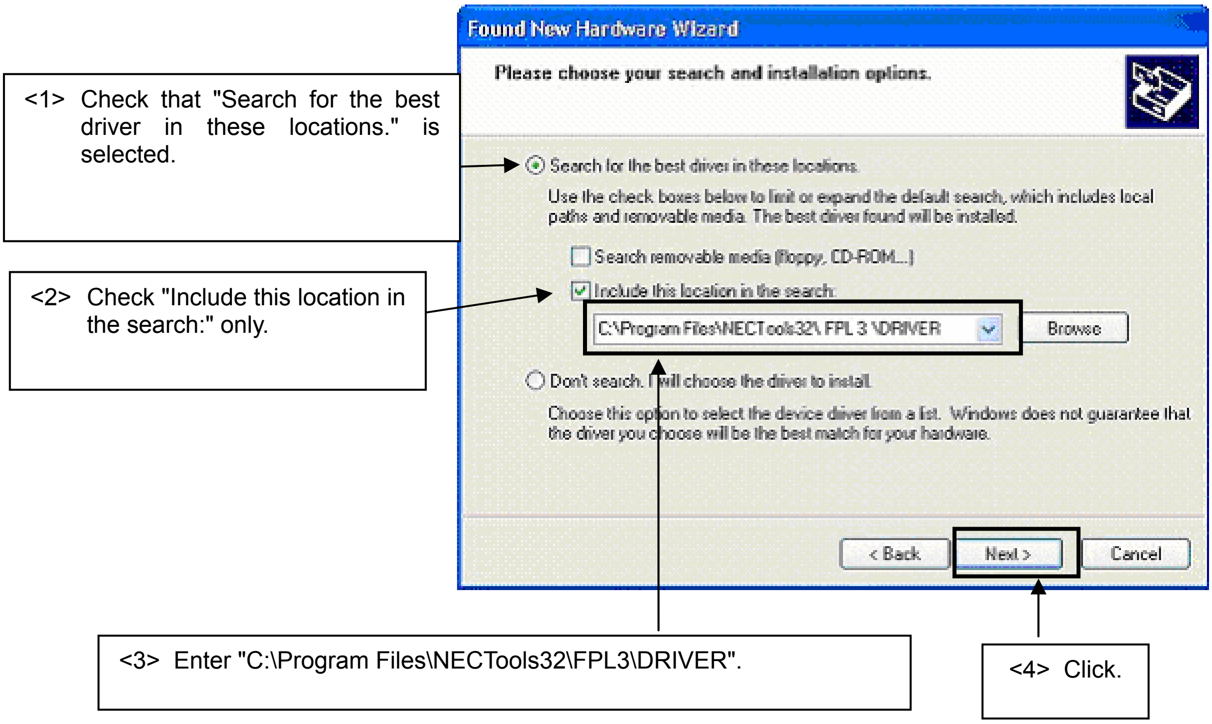
5. Proceed to the installation of the USB Serial Port driver. Click Next>.

Figure 27: Found New Hardware Wizard 2 (Windows XP)



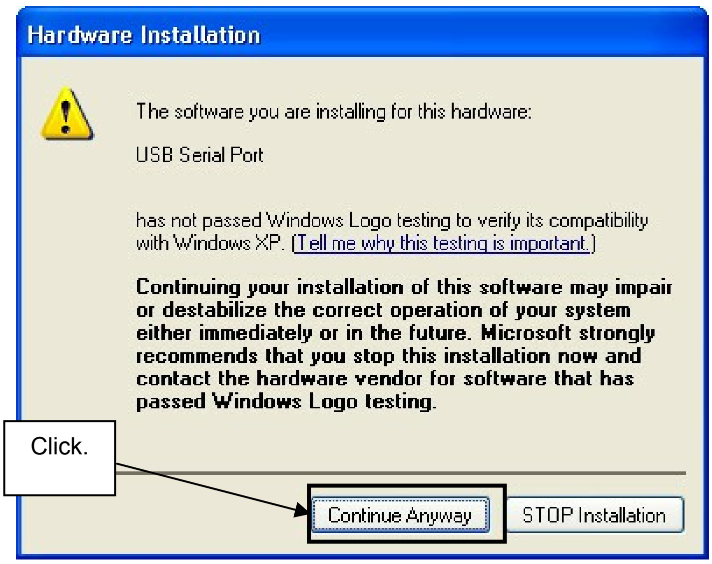
6. Check that "Search for the best driver in these locations." is selected. Check the "Include this location in the search:" check box and enter "C:\Program Files\NECTools32\FPL3\DRIVER", then click Next>.

Figure 28: Search Location Specification 2 (Windows XP)



7. As shown below, "has not passed Windows Logo testing to verify its compatibility with Windows XP." is displayed. Click Continue Anyway.

Figure 29: Windows XP Logo Testing 2 (Windows XP)



8. When the window below is displayed, the installation of the USB driver is completed. Click Finish.

Figure 30: USB Serial Port2 Driver Installation Completion (Windows XP)

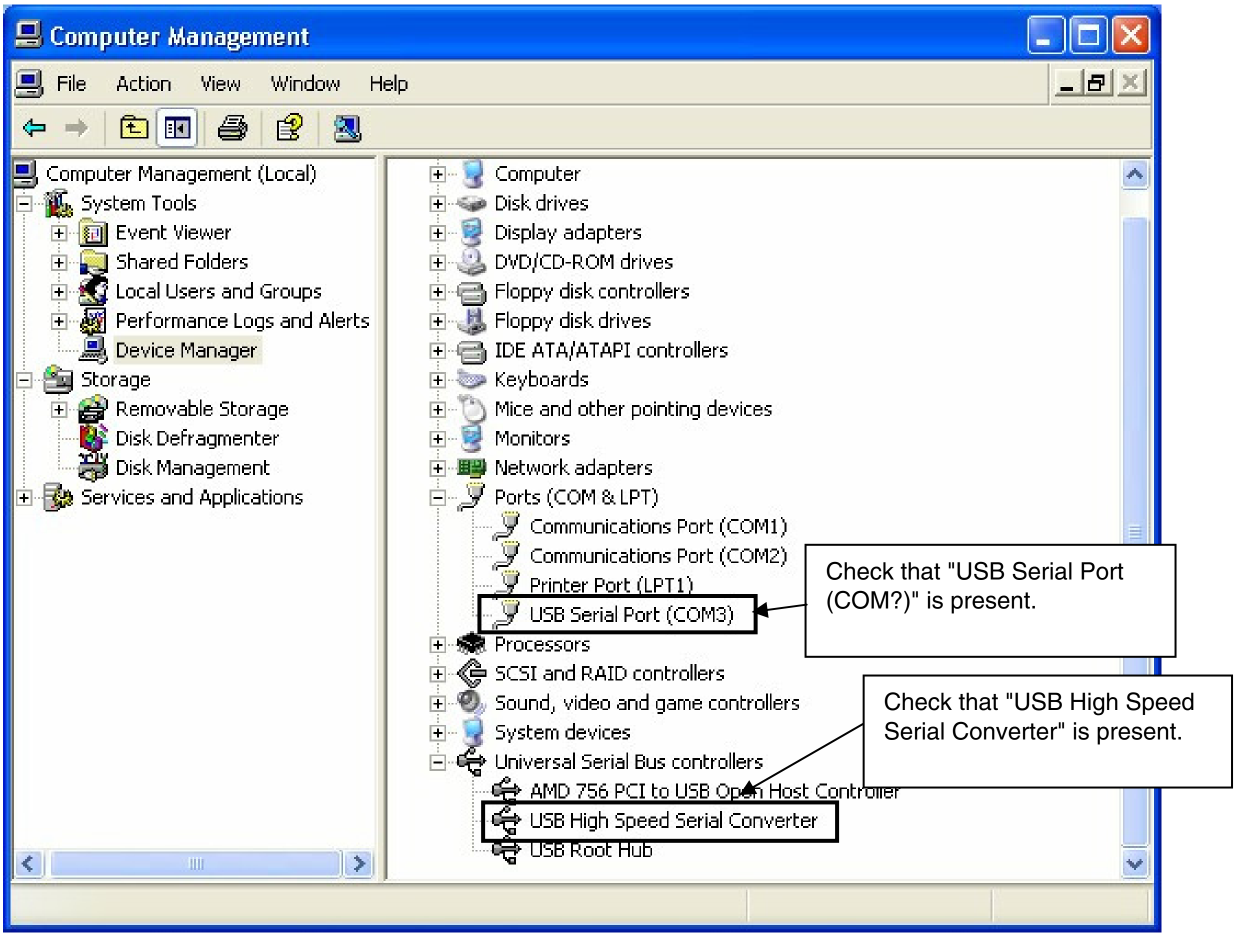


8.6 Confirmation of USB Driver Installation

After installing the two types of drivers, check that the drivers have been installed normally, according to the procedure below. When using the *78K0/Fx2 – CAN it!* board in combination with FPL3 GUI, the information to be checked here is needed.

By clicking the "Device Manager" tab, check that the drivers are installed normally.

Figure 31: Device Manager



For Windows 98SE/Me

Caution Do not select Update and Erase when communicating with the target device.

For Windows 2000/XP

Caution Do not perform "Hardware Modification Scan" when communicating with the target device.

Remark In the GUI port list box, the same communication port as COM? of USB Serial Port (COM?) needs to be selected.

If the drivers above are not displayed, or the mark "×" or "!" is prefixed, refer to **CHAPTER 11 TROUBLESHOOTING**.

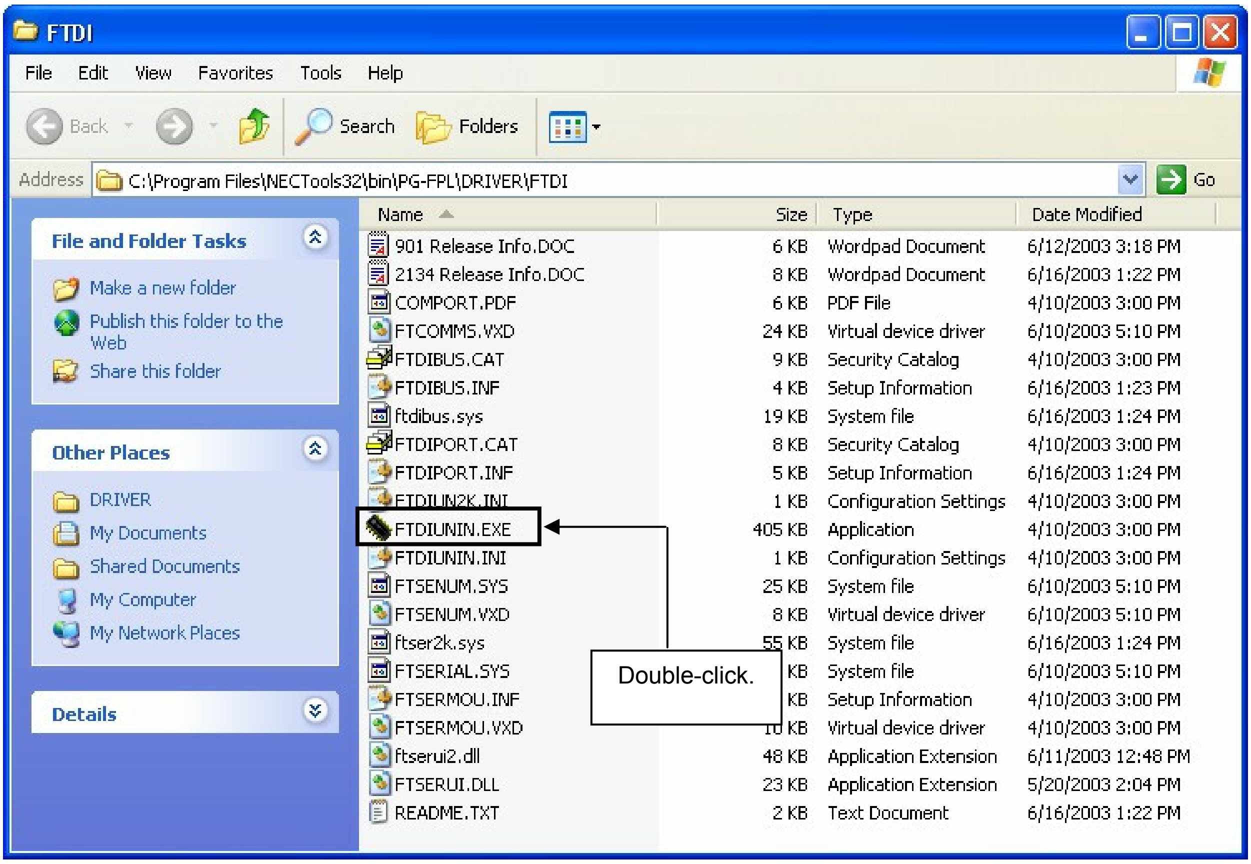
8.7 Driver deinstallation

The driver deinstallation program is installed on the host machine when the FPL3 software is installed. Use the procedure below for deinstall the USB driver.

1. When using Windows XP, log on as the computer administrator. When using Windows 2000, log on as the Administrator.

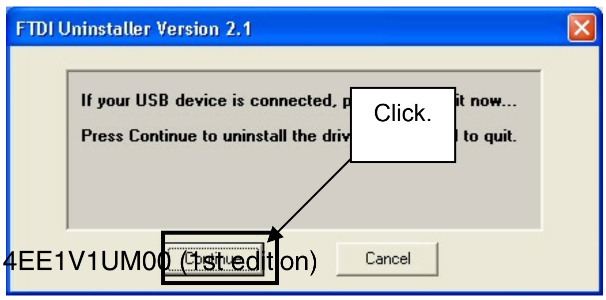
2. Double-click in the order from "My Computer" to "(C:)" to "Program Files" to "NECTools32" to "FPL3" to "DRIVER". "Ftdiunin.exe" is displayed. Double-click "Ftdiunin.exe".

Figure 32: Driver Uninstallation



3. Click Continue.

Figure 33: Driver Uninstaller



4. Click Finish to complete driver uninstallation.

Figure 34: Completion of Driver Uninstallation



Caution	If the GUI software is uninstalled earlier, "Ftdiunin.exe" is also deleted. At this time, delete "USB Serial Port (COM?)" and "USB High Speed Serial Converter" from Device Manager manually.

9. FPL3 FLASH programming software

9.1 Introduction

The parameter file for the 78K0/FF2 device is installed automatically during installation of FPL3 GUI, folder <FPL3 install-path>\PRM. Nevertheless, newest version of parameter file for the µPD78F0893 device can by download from the NEC Electronics Web site.

Download the parameter file for the PG-FP4 from the following NEC Electronics Web site:

http://www.eu.necel.com/updates

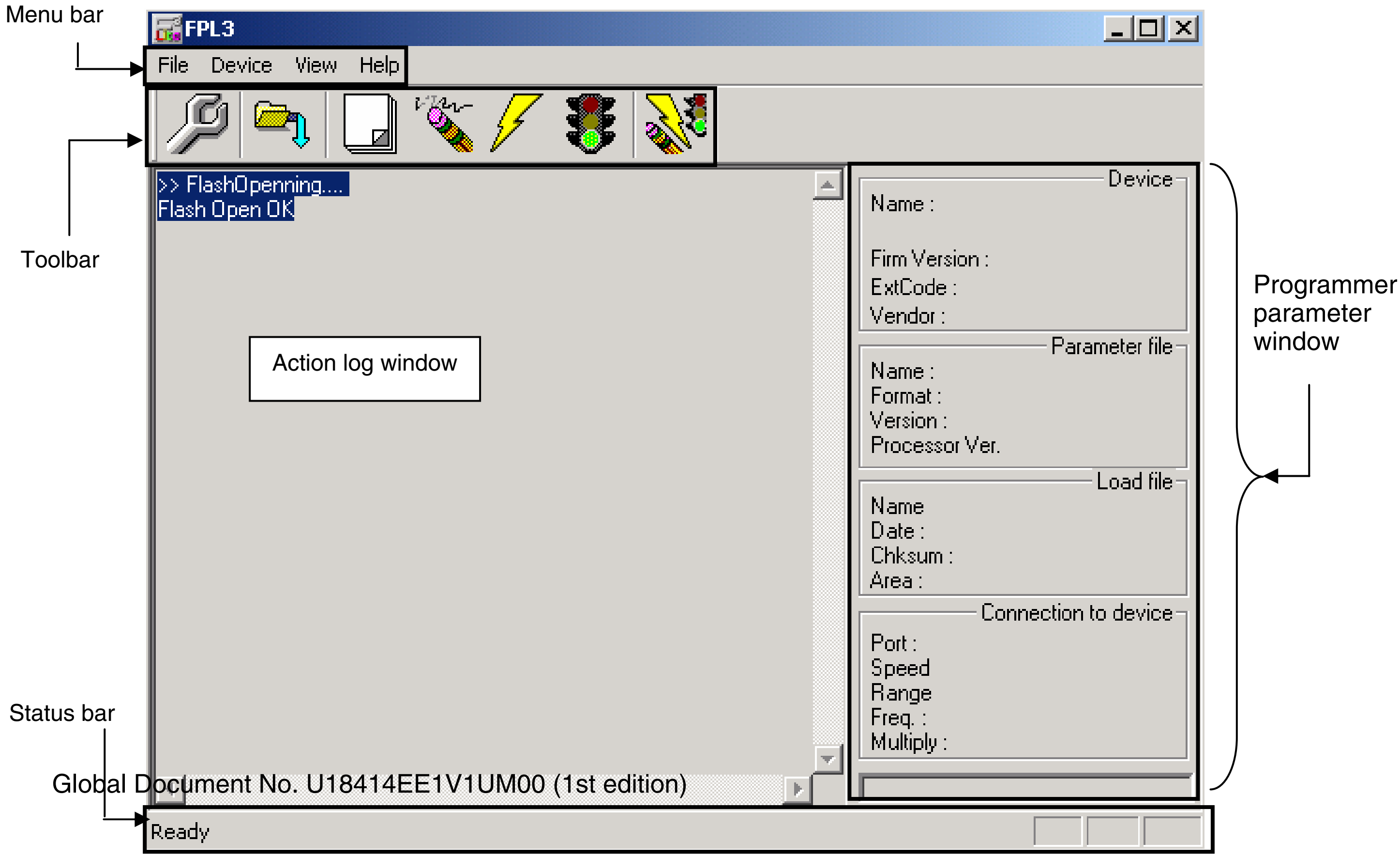
Copy the parameter file downloaded from the NEC Electronics Web site into sub-directory <FPL3.EXE-install-path>\PRM created during GUI software setup (refer to **CHAPTER 8 SOFTWARE INSTALLATION**)..

9.2 Starting up the GUI Software

- GUI software startup
  Select FPL3.EXE from the start menu to start the FPL3 GUI software.

  When the GUI software is started normally, the following screen appears.

Figure 35: GUI Software Main Window



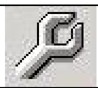
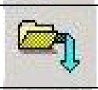
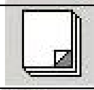
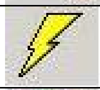
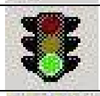
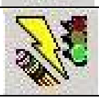
This window consists of the following items:

Name	Display Information
Menu bar (displayed at the top)	Displays menu items executable by the FPL3.
Toolbar (displayed under the menu bar)	Displays frequently used commands as icons.
Action log window (displayed under the toolbar)	Displays an FPL3 action log.
Programmer parameter window (displayed to the right of the action log window)	Displays programming parameter settings.
Status bar	Displays status.

9.3 Toolbar

The toolbar contains buttons for starting the important procedures of the FPL3.

Figure 36: Toolbar Buttons

	[Device] → [Setup] button
	[File] → [Load] button
	[Device] → [Blank Check] button
	[Device] → [Erase] button
	[Device] → [Program] button
	[Device] → [Verify] button
	[Device] → [Autoprocedure(EPV)] button

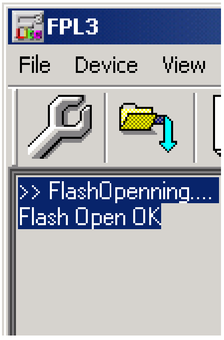
9.4 Menu Bar

Depending on the actual device status and device type, some menu items may be enabled or disabled.

9.4.1 [File] menu

Clicking the [File] menu displays the pull-down menu as shown below.
This menu mainly contains commands related to file operation.

Figure 37: [File] Menu

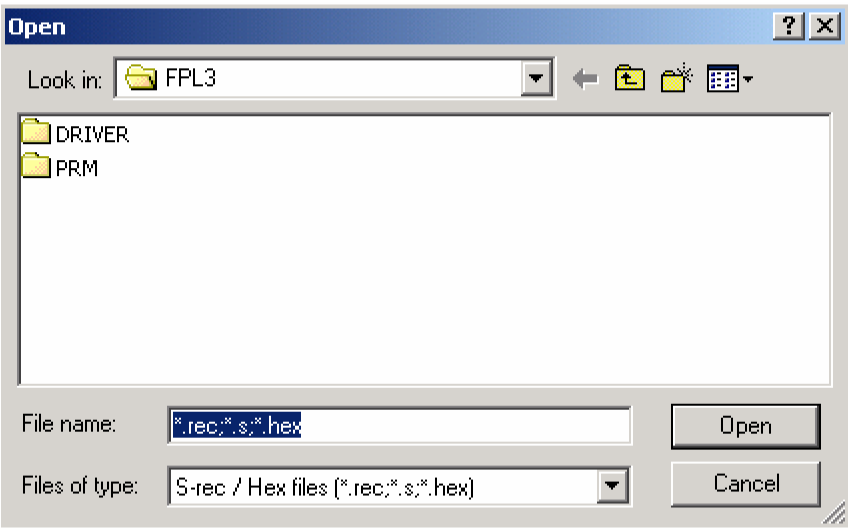


(1) [Load] command



The [Load] command allows you to select a program file.
The selected program file is programmed into the flash memory of the device by executing the [Program] command or [Autoprocedure(EPV)] command.

Figure 38: HEX File Selection Window



The file selection window for program loading displays the most recently used directory to which a user program has been loaded. After a user program is loaded, a checksum calculation is made and the result is displayed in the programmer parameter window.

[Open] button]
Selects a user program as a program to be written to the target device.

[Cancel] button]
Closes the window without selecting a program.

(2) [Quit] command

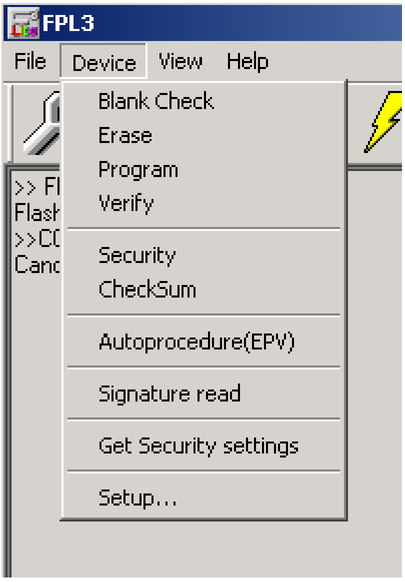
The [Quit] menu is the command for terminating the FPL3 GUI software. Clicking × on the right side of the task bar also terminates the FPL3 GUI software.
User settings are saved in the FPL3.INI[Note] file, so that the GUI software starts up next time with the same settings.

Note FPL3.INI is created in the Windows folder when Windows 98SE, Windows Me, or Windows XP is used.
When Windows 2000 is used, FPL3.INI is created in the Winnt folder.

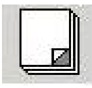
9.4.2 [Device] menu

Clicking the [Device] menu displays the pull-down menu as shown below.
This menu mainly contains commands for programming operations such as deletion, programming, and verification on the target device.

Figure 39: [Device] Menu



(1) [Blank Check] command



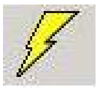
The [Blank Check] command allows you to make a blank check on the target device connected to the FPL3. If the flash memory of the target device is erased, a blank check is terminated normally. If the flash memory is not completely erased, the indication "not blank" is provided. Before starting programming, erase the flash memory of the target device.

(2) [Erase] command



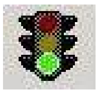
The [Erase] command erases the flash memory of the target device connected to the FPL3. While the flash memory is being erased, the progress status is displayed in the action log window to indicate programmer operation.
The execution on the [Blank Check] command before the [Erase] command is executed follows the setting of 'Command options' of the Advance tab displayed by selecting [Device] → [Setup].
Upon completion of [Erase] command execution, the GUI software displays the result of executing the command on the target device.

(3) [Program] command



The [Program] command sends a specified user program to the target device and writes the program to the flash memory.
The execution of Verify operation for detecting an error in user program communication from the FPL3 to the target device after the execution of the [Program] command follows the setting of the 'Command options' on the Advance tab displayed by selecting [Device] → [Setup].
During programming, the progress status is displayed in the action log window to indicate programmer operation. This progress status display window displays the progress status on target device programming by percentage.
Upon completion of [Program] command execution, the GUI software displays the result of executing the command on the target device.

(4) [Verify] command



The [Verify] command sends a specified user program to the target device connected with the FPL3, and performs verification against the data written to the flash memory of the target device.
During verification, the progress status is displayed in the action log window to indicate programmer operation. This progress status display window displays the progress status of target device verification by percentage.
Upon completion of [Verify] command execution, the GUI software displays the result of executing the command on the target device.

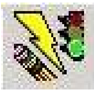
(5) [Security] command

The [Security] command initiates the programming of the security flag of the target device connected to the FPL3. Set ‘Security flag settings’ on the Advance tab of the [Device] > [Setup…] menu.

(6) [Checksum] command

The [Checksum] command reads the checksum value of the target device connected with the FPL3.
This value differs from the value displayed in the parameter window of the main window.

(7) [Autoprocedure(EPV)] command

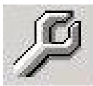


The [Autoprocedure(EPV)] command executes the [Erase] command, [Program] command and [Verify] command in succession.
When a user program is to be resent to the target device for comparison with the data written to the flash memory of the target device because of a user program communication error, execute the [Program] command by selecting [Device] → [Setup] and specifying 'Command options' on the Advance tab, then set the automatic execution of the [Verify] command.
During EPV execution, the progress status is displayed in the action log window to indicate programmer operation. For a selected command, its execution operation, and messages, refer to **CHAPTER 10 HOW TO USE FPL3**.
Upon completion of [Autoprocedure(EPV)] command execution, the GUI software displays the result of executing the command on the target device.

(8) [Signature read] command

The [Signature read] command reads the signature information (device name, flash memory information, and so forth) of the target.

(9) [Setup] command

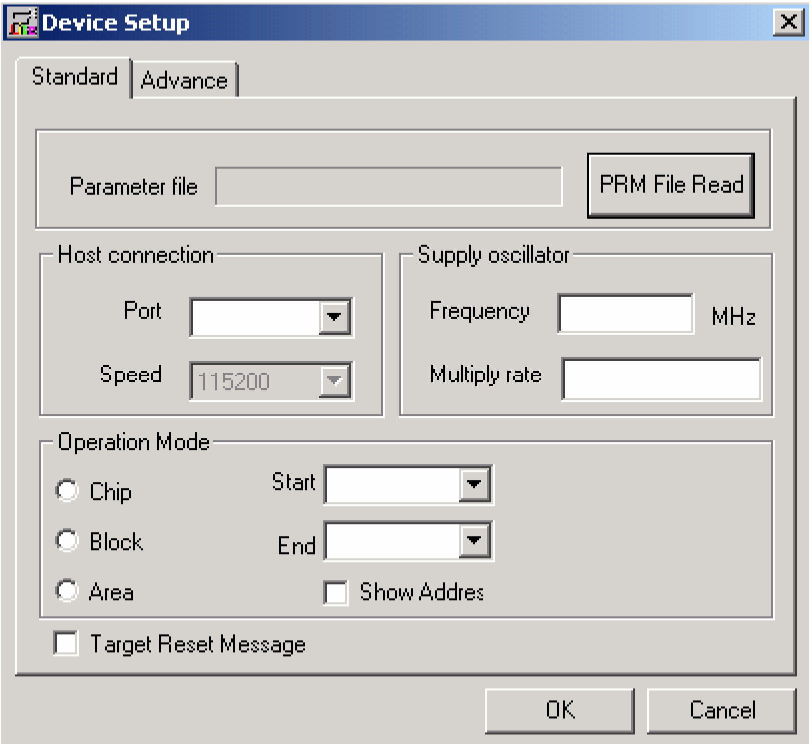


The [Setup] menu allows you to make settings related to flash memory rewriting according to the user environment and to set command options. Each time the GUI software is started, the most recently used parameter file (.PRM) is read and the settings are displayed. The [Setup] menu allows you to modify the settings of items other than those items consisting of shadowed characters according to the user environment.

(a) Standard setup

This menu is used to set the environment for rewriting the flash memory of the target device.
The mode of communication with the target, the operating clock, and so forth differ depending on the device used. For details, refer to the manual of the device used, when making settings.
The window shown below is opened.

Figure 40: Device Setup Window - Standard



This window shows all basic options that can be set in accordance with the user environment and target device.

[OK button]
Clicking the OK button saves the settings on the Standard and Advance menus and closes the window.

[Cancel button]
Clicking the Cancel button closes the window without saving the settings on the Standard and Advance menus.

<1> Parameter file

This file holds parameters and timing data required to rewrite the flash memory of the target device. Do not modify the data in the parameter file because the data is related to the guarantee of rewrite data.

The parameter file is protected by the checksum function. If the checksum result indicates an error, the FPL3 does not accept the parameter file.

Figure 41: Device Setup Window - Parameter File Selection

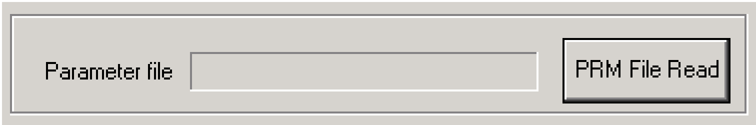
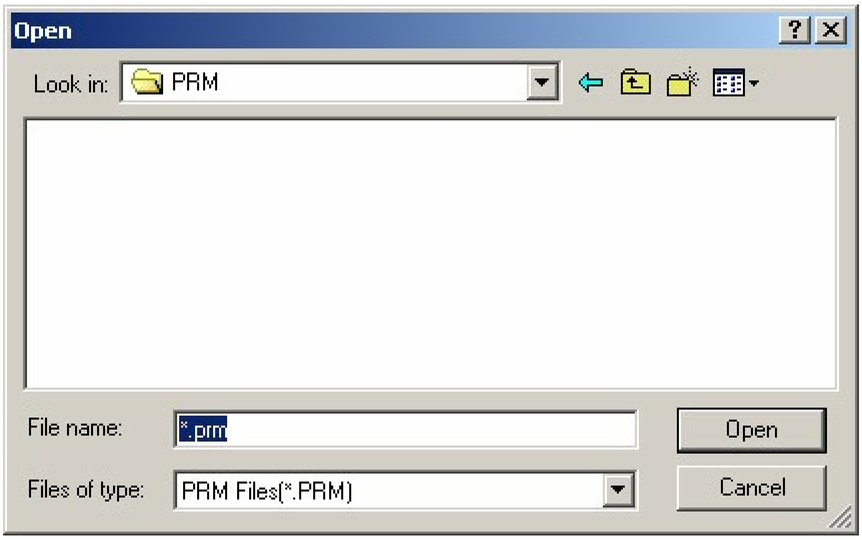


Figure 42: Parameter File Selection Window



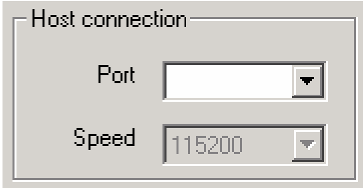
[PRM File Read button]

A window for specifying a parameter file is displayed. Specify a desired file then click Open.

<2> Host connection

"Host connection" is used to select a channel for communication between the *78K0/Fx2 – CAN it!* board and host machine.

Figure 43: Device Setup Window – Host connection



[Port list box]

Select a channel for communication between the *78K0/Fx2 – CAN it!* board and host machine.

- COM1 to COM256

Remark Selectable ports can be checked using Device Manager. For details, refer to **CHAPTER 8.6 Confirmation of USB Driver Installation**.

[Speed list box]

Select a communication rate for the selected communication channel from the following:

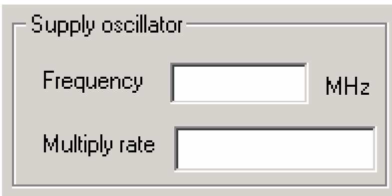
- 9600 bps
- 19200 bps
- 38400 bps
- 115200 bps

Remark For selectable communication rates, refer to the user's manual of the device used.

<3> Supply oscillator

"Supply oscillator" is used to select a clock that determines programming, data transfer, and a transfer rate.

Figure 44: Device Setup Window - Supply Oscillator Selection



[Frequency box]

Sets the clock frequency of the target system.

The range of operating frequency varies from one device to another. So, check the specifications of the device used before making a setting.

[Multiply rate]

Specifies the division rate or multiplication rate of the target device.

If the target device has an on-chip PLL circuit, enter a division rate or multiplication rate according to the use environment.

The selectable division rate or multiplication rate differs depending on the device. Check the specifications of the device used before making a setting.

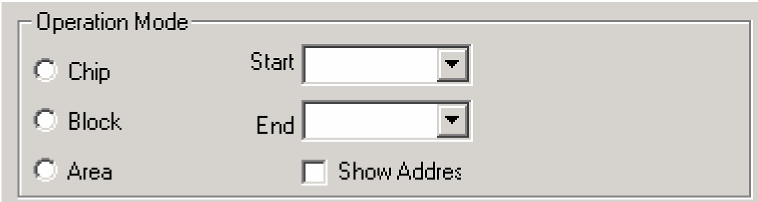
If the target device does not have an on-chip PLL circuit, select "1.0".

On the initial screen, the default setting is displayed according to the parameter file.

<4> Operation Mode

The setting of "Operation Mode" may divide the flash memory of some target devices into blocks or areas.
This menu is used to select an operation mode of the flash memory. Some devices do not have the block and area division modes, and some devices have only one of the modes. In these cases, a nonexisting mode is unchoosable.

Figure 45: Device Setup Window - Operation Mode



[When Chip is selected]
The entire flash memory area of the target device is subject to rewrite processing.

[When Block is selected]
Specify the Block number range subject to rewrite processing by using Start/End.
The Start/End list boxes display the Block numbers where the flash memory of the target device is configured.

[When Area is selected]
Specify the Area number range subject to rewrite processing by using Start/End.
The Start/End list boxes display the Area numbers where the flash memory of the target device is configured.

[Show Address check box]
Specify whether numbers or addresses are displayed in the Start/End list boxes.
If this check box is checked, addresses are displayed.
If this check box is not checked, numbers are displayed.

<5> Target Reset Message

By checking the Target Reset Message check box, the window promoting the reset operation manually is displayed even when the reset signal cannot be connected to the target cable.

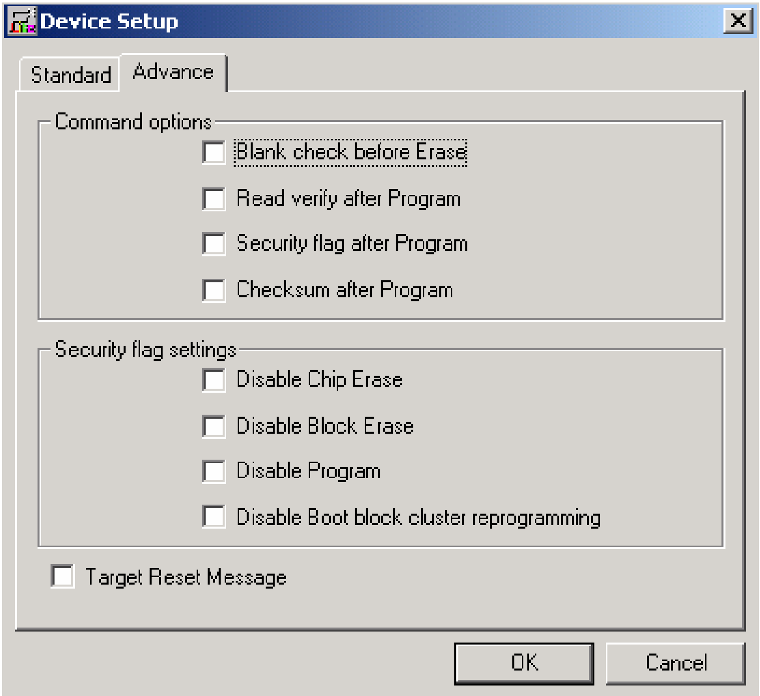
Figure 46: Device Setup Window – Target Reset Message



(b) Advance setup

The Advance setup menu is used to specify the command options and security flag settings.
When "Advance" is clicked, the following window is displayed:

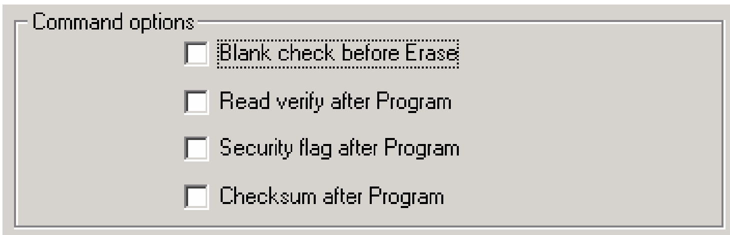
Figure 47: Device Setup Window - Advance



<1> Command options

This dialog box is used to specify the FPL3 flash processing command options.

Figure 48: Device Setup Window - Command options



[Blank check before Erase check box]
If this check box is checked, blank check is made before the Erase command or EPV command is executed.
If the result of a blank check indicates OK, erase processing is not executed.

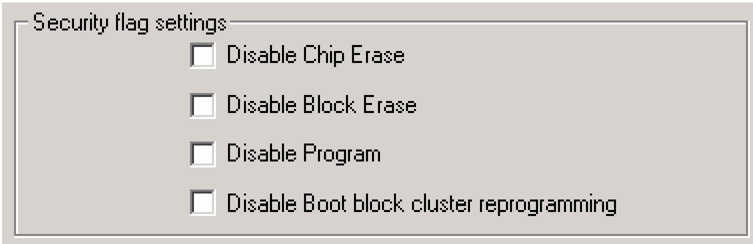
[Read verify after Program check box]
If this check box is checked, write data is sent from the programmer after execution of the Program command and EPV command, then the data is verified against the data written to the flash memory.

[Security flag after Program check box]
If this check box is checked, automatic programming of the selected security flag is executed after execution of the [Program] and [Autoprocedure (EPV)] commands.

[Checksum after Program check box]
If this check box is checked, the flash memory checksum value of the target device is read from the target device after execution of the Program command and EPV command.
This value differs from the value displayed in the parameter window of the main window.

<2> Security flag settings
The <Security flag settings> dialog box is used to specify which security function is valid.

Figure 49: Device Setup Window – Security flag settings



Caution: The following is the correspondence between the [Erase] and [Program] Commands when the security functions of a 78K0/FF2 microcontroller are valid!

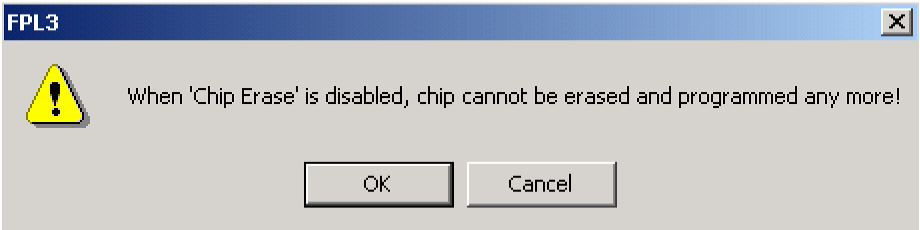
	Chip Erase Command	Block Erase Command	Program Command
Disable Chip Erase	Invalid	Invalid	Valid **(Note1)**
Disable Block Erase	Valid	Invalid	Valid
Disable Program	Valid	Invalid	Invalid
Disable Boot block cluster reprogramming	Invalid	Valid **(Note2)**	Valid **(Note2)**

Notes 1: Since the [Erase] command is invalid, the data that differs from the data already written in the flash memory cannot be written.

Notes 2: Valid only for the area other than the area specified as the boot area

[Disable Chip Erase check box]
If this check box is checked, the [Erase] command becomes invalid in the entire flash memory area of the target device.
At this time, the warning message shown below is displayed.

Figure 50: Device Setup Window – Disable Chip Erase



Caution: Be aware that if the security flag is set in the target device, neither erasing nor writing to the device can be enabled afterward!!!

[Disable Block Erase check box]
If this check box is checked, the [Erase] command becomes invalid in all the blocks of the flash memory selected under Operation Mode in the Standard Setup menu. This setting is cleared by the [Erase] command when Chip was selected under Operation Mode.

[Disable Program check box]
If this check box is checked, the [Program] command becomes invalid, as does the [Erase] command in all the blocks of the flash memory selected under Operation Mode in the Standard Setup menu.
The [Erase] command for the entire flash memory area is valid. This setting is cleared by the [Erase] command when Chip was selected under the Operation Mode.

[Disable Boot block cluster reprogramming check box]
If this check box is checked, the boot area is set with the boot block set in the Boot block Cluster setting as the last block. At this time, the warning message shown below is displayed.

Figure 51: Device Setup Window – Disable Boot Cluster Reprogramming warning

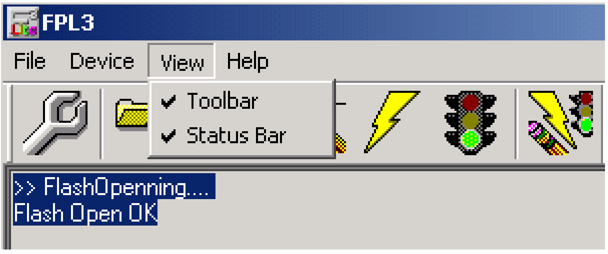


Caution: Be aware that if the security flag is set in the target device, the boot area cannot be rewritten to the device afterward!!!

9.4.3 [View] menu

Clicking the [View] menu displays the pull-down menu shown below.
This menu contains commands for setting whether to display the toolbar and status bar.

Figure 52: [View] Menu



(1) [Toolbar] command

Checking the [Toolbar] command displays the toolbar. Unchecking the command hides the toolbar.

(2) [Status Bar] command

Checking the [Status Bar] command displays the status bar. Unchecking the command hides the status bar.

9.4.4 [Help] menu

Clicking the [Help] menu displays the following pull-down menu:

Figure 53: [Help] Menu



(1) [About FPL3] command

The [About FPL3] command opens the program entry window as shown below and indicates the version.

Clicking OK terminates the display.

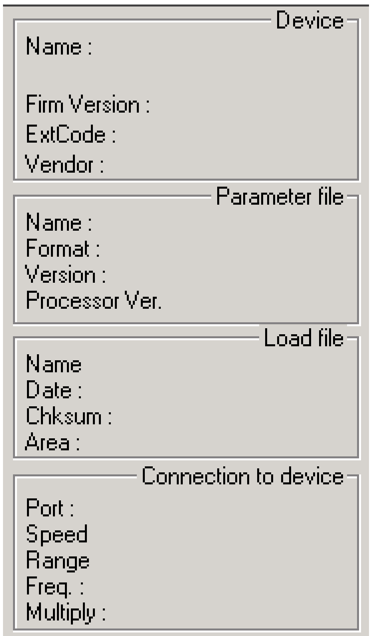
Figure 54: About FPL3 Window



9.5 Programmer Parameter Window

This window displays the settings of the programming parameters.

Figure 55: Programmer Parameter Window



[Device]
Updated after communication with the target device to display information about the target device.

[Parameter file]
Updated after [Setup] command execution to display information about a read parameter file.

[Load file]
Updated after [Load] command execution to select information about a selected program file.

[Connection to device]
Updated after [Setup] command execution to display information about the connection with the target device.

10. How to use FPL3 FLASH programming software

This chapter explains the basic operation of the FPL3 GUI for programming the *78K0/Fx2 – CAN it!* board. This chapter covers how to start the system, execute the EPV command, and program the 78K0/FF2 target device.

The conditions of the series of operations described in this chapter are as follows:

Hardware configuration of *78K0/Fx2 – CAN it!* :

Base board :	*78K0/Fx2 – CAN it!*
Target device :	78K0/FF2 (µPD78F0893)
Clock :	12 MHz
Voltage level :	5 V

Software configuration of FPL3:

Parameter file:	78F0893.PRM
Clock setting :	12 MHz Multiplied by 1
Port :	COM4 (115200 bps)
Operation mode:	Chip
Write HEX :	ADC_demo.hex
Option setting :	Blank check before Erase

(1) Installing the FPL3 GUI software

Install the FPL3 GUI software on the host machine you are using, by referring to **CHAPTER 8 SOFTWARE INSTALLATION** (if the software has not been installed yet).

(2) Installing the driver

Install the USB driver on the host machine you are using, by referring to **CHAPTER 8 SOFTWARE INSTALLATION** (if the driver has not been installed yet).

(3) Installing the parameter file

The parameter file for the 78K0/FF2 device is installed automatically during installation of FPL3 GUI, folder <FPL3 install-path>\PRM. Nevertheless, newest version of parameter file for the µPD78F0893 device can by download from the NEC Electronics Web site.

Download the parameter file for the PG-FP4 from the following NEC Electronics Web site:

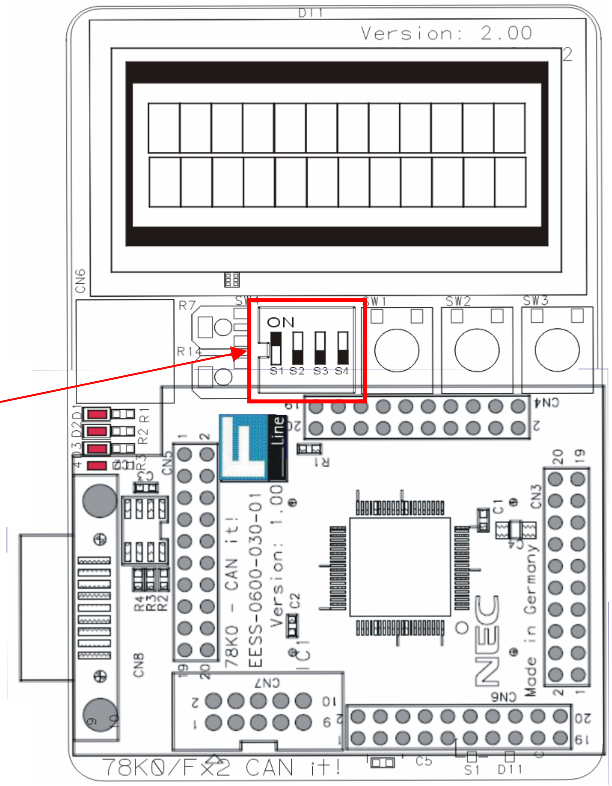
http://www.eu.necel.com/updates

Copy the parameter file downloaded from the NEC Electronics Web site into sub-directory <FPL3.EXE-install-path>\PRM created during GUI software setup (refer to **CHAPTER 8 SOFTWARE INSTALLATION**).

(4) Connecting and starting

<1> Set the *78K0/Fx2 – CAN it!* board to the FLASH programming mode by switching SW4/S1 to ON:

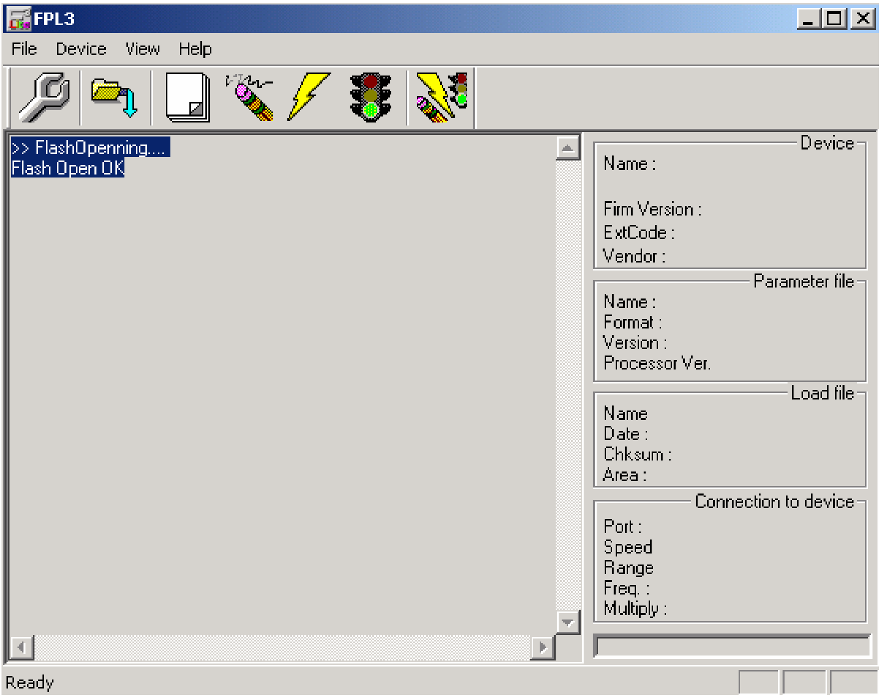
SW4	Setting
S1	ON
S2	OFF
S3	OFF
S4	OFF



<2> <Plug and Play> Connect the *78K0/Fx2 – CAN it!* board with the host machine via the USB cable. If the connection was already done, press the reset button SW1 to release the FLASH programming mode.

<3> Start the FPL3 GUI.

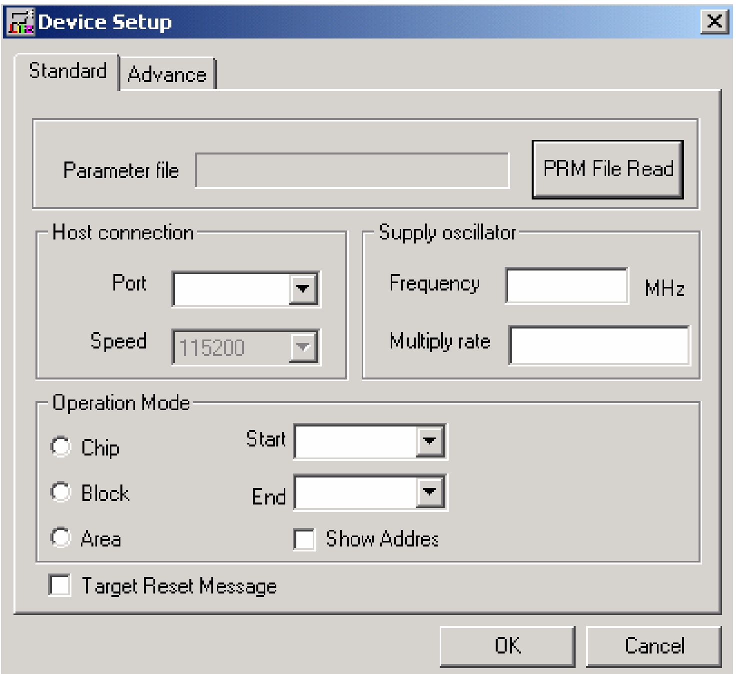
Figure 56: GUI Software Startup Screen



(5) Setting the programming environment

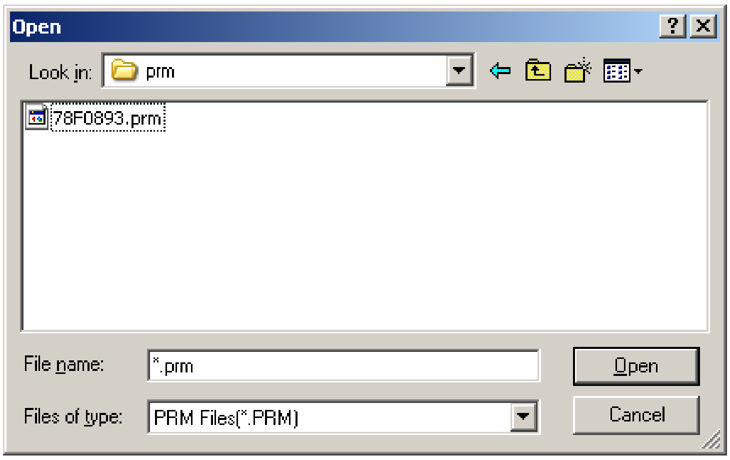
<1> Select [Device] → [Setup] from the menu bar.
<2> The Standard dialog box for device setup is activated.

Figure 57: <Standard Device Setup> Dialog Box



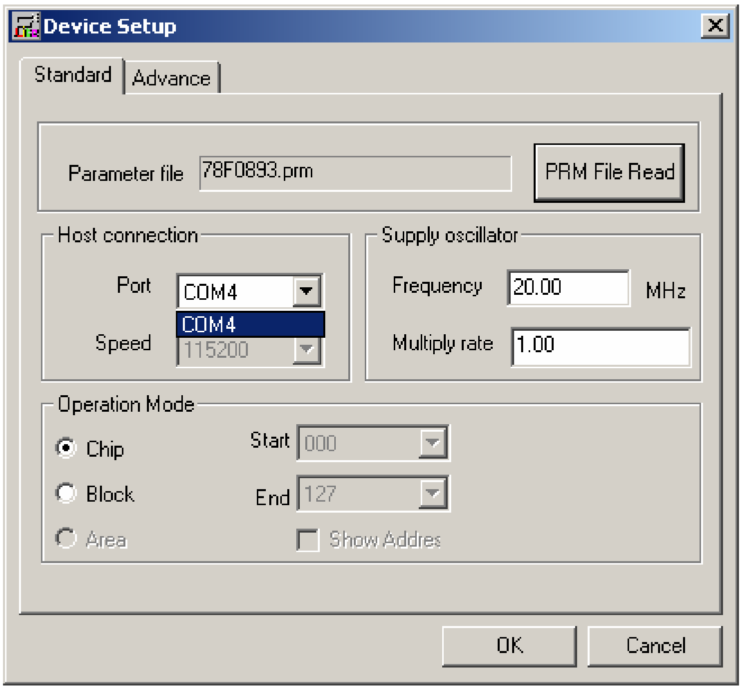
<3> Click PRM File Read to open the parameter file selection window.
Select the parameter file "78F0893.prm" then click Open.

Figure 58: Parameter File Selection



<4> From the Port list box, select the communication port that matches the host machine being used. Select the communication speed of the Host connection.

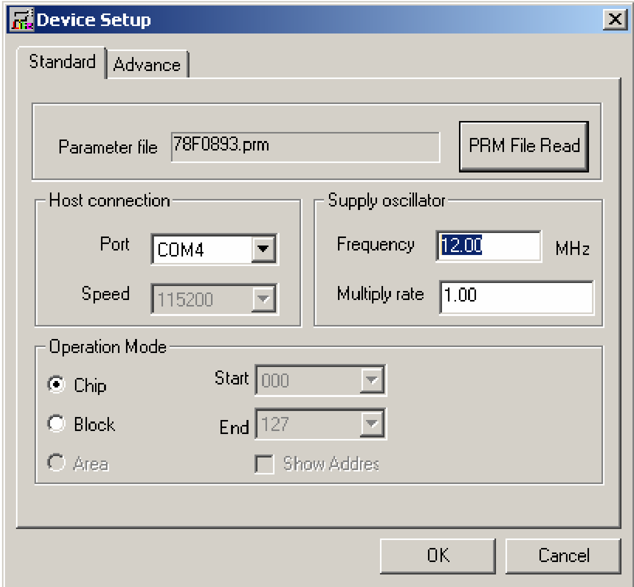
Figure 59: Port Selection



Remark Selectable ports can be checked using Device Manager. For details, refer to **CHAPTER 8.6 Confirmation of USB Driver Installation**.

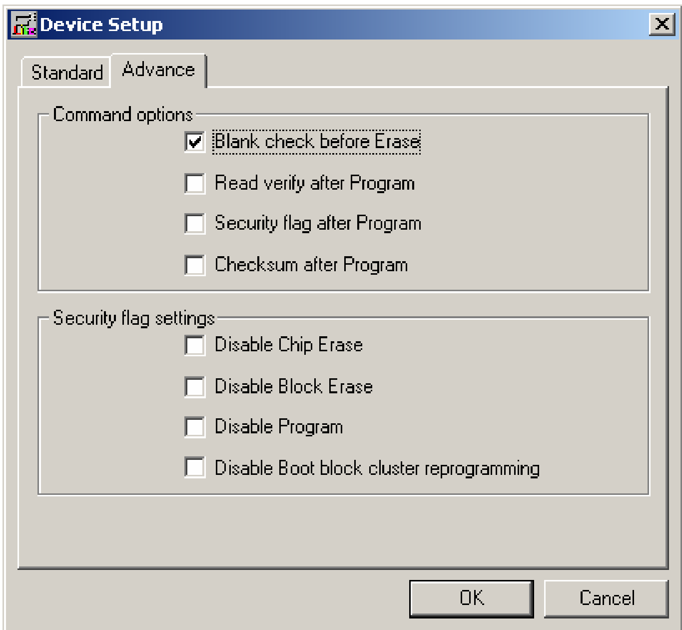
<5> Set "Supply oscillator" according to the specifications of the *78K0/Fx2 – CAN it!* board, “Frequency = 12.00 MHz” and “Multiply rate = 1.00”. In "Operation Mode", please specify the “Chip” mode. The following figure shows the recommended settings:

Figure 60: <Standard Device Setup> Dialog Box after Setting



<6> Switch to the Advance dialog box.

Figure 61: <Advance Device Setup> Dialog Box

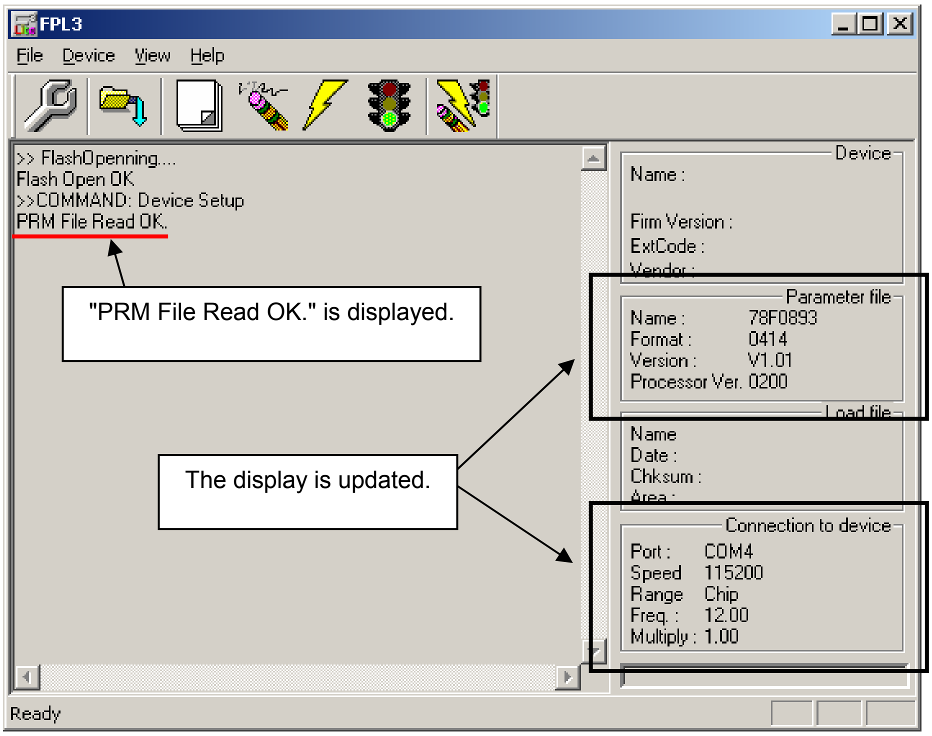


<Command options>

Blank check before Erase : Checked

<7> Click the OK button. The GUI software sets the parameters.
When the settings have been completed, the following screen is displayed:

Figure 62: Completion of Parameter Setting

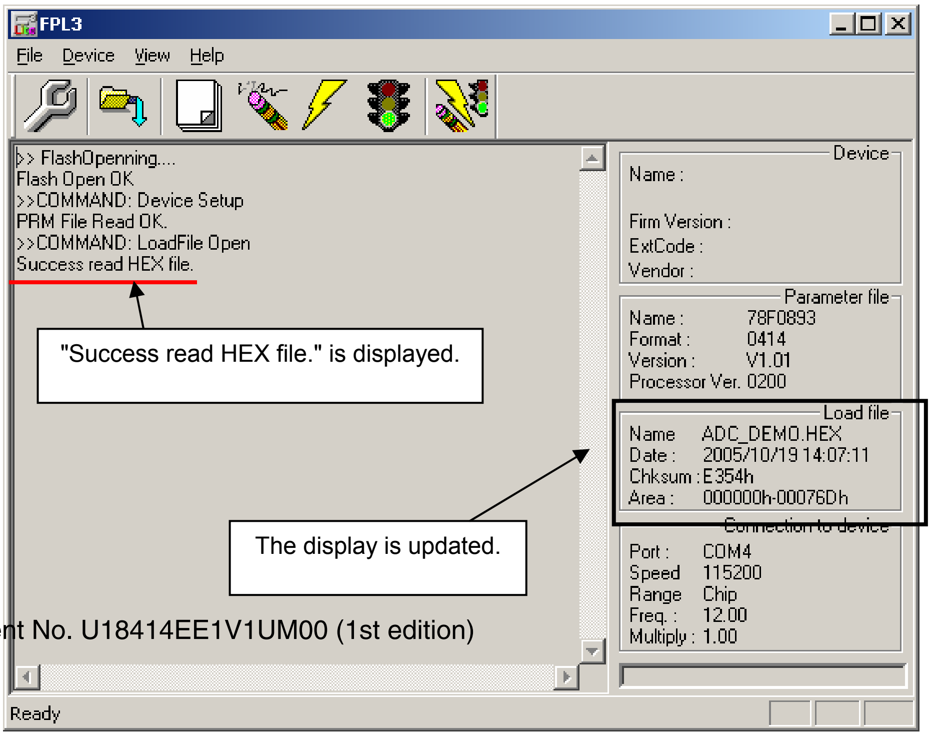


(6) Selecting a user program

<1> Select [File] → [Load].

<2> Select a program file to be written to the target device, then click Open.

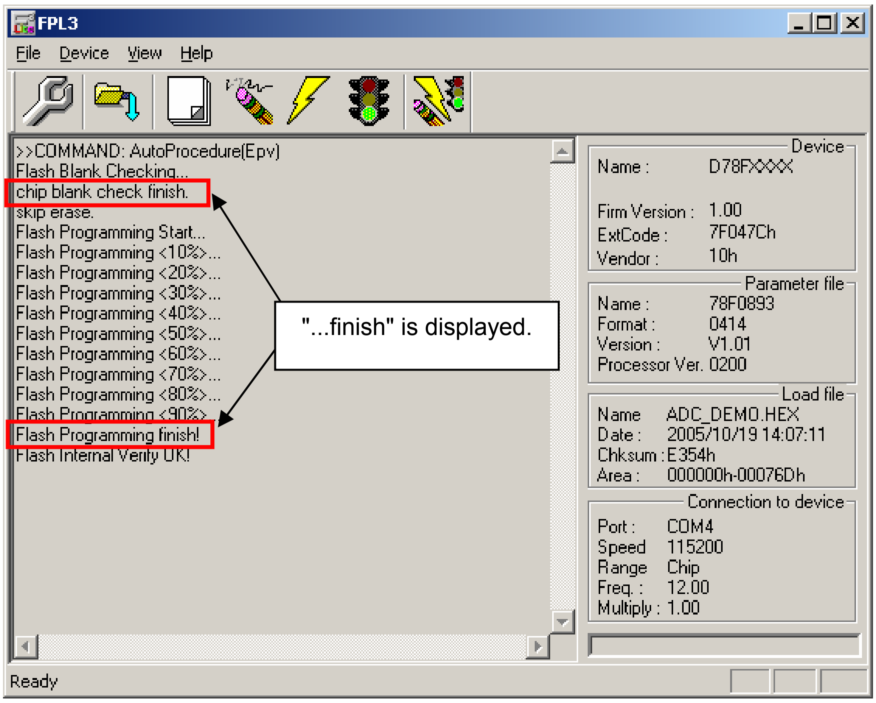
Figure 63: After Downloading



(7) [Autoprocedure(EPV)] command execution

Select [Device] → [Autoprocedure(EPV)] from the menu bar.
When the [Autoprocedure(EPV)] command is executed, Blank Check → Erase → Program and FLASH Internal Verify are executed sequentially for the μPD78F0893 device.

Figure 64: After EPV Execution



(8) Terminating the GUI

Select [File] → [Quit] to terminate the GUI software. All settings executed so far are saved in the FPL3.INI file, so that those settings can be reused when the GUI software is restarted.

(9) Execute "ADC_DEMO" application

Set the *78K0/Fx2 – CAN it!* board to the normal operation mode by switching SW4/S1 to OFF.
< Plug and Play> the *78K0/Fx2 – CAN it!* board to start in normal operation mode or press the reset button SW1 to release the normal operation mode.

(10) Restarting the GUI

When the system is restarted, the same screen as shown in Figure 62 appears.

11. TROUBLESHOOTING

In driver installation, recognition based on Plug and Play is disabled.

Cause:
The USB connector may not be inserted normally into the USB port of the personal computer.
Action:
Check that the USB connector is inserted fully into the USB port of the personal computer. Alternatively, disconnect the USB connector, then insert the USB connector again after a while.

The driver file cannot be found at a specified location.

Cause:
The FPL3 FLASH programming software may not be installed correctly.
Action:
Install the GUI software again by referring to **CHAPTER 8 Software Installation**.

In checking by Device Manager, "USB Serial Port" or "USB High Speed Serial Converter" is not displayed. Alternatively, the "!" or "×" is prefixed.

Cause:
The USB connector may not be inserted normally into the USB port of the personal computer.
Action:
Check that the USB connector is inserted fully into the USB port of the personal computer. Alternatively, disconnect the USB connector from the USB port, then insert the USB connector again after a while.

Cause:
The driver may not be installed correctly.
Action:
<1> When this product is connected to the personal computer, right-click the driver marked with "!" or "×".
Click Erase when displayed.
<2> On Device Manager, execute [Hardware Modification Scan].
<3> Install the driver again with Plug and Play.

Cause:
The device may not be recognized (in the case of connection with the USB hub).
Action:
Try the following:

- Disconnect the USB connector, then insert the USB connector again.
- Connect the USB connector to another port of the USB hub.

If the same symptom occurs, do not use the USB hub, but directly connect the connector to the USB port of the personal computer.

When this product is connected with a personal computer, the "Add New Hardware Wizard" screen is displayed.

Cause:
If the USB connector of this product is not inserted into the USB port used at the installation time but into another USB port, this product may be recognized as a new hardware item.
Action:
Install the driver by referring to **CHAPTER 8.5 USB Driver Installation**.

Communication with the *78K0/Fx2 – CAN it!* board is disabled.

Cause:
The driver may not be installed correctly.
Action:
Check if "USB Serial Port" and "USB High Speed Serial Converter" are installed correctly by referring to **CHAPTER 8.5 USB Driver Installation**.

Cause:
The COM port selected via the “Port list box” within device setup menu of FPL3 may not be set correctly.
Action:
Set the port checked using Device Manager.

Cause:
The *78K0/Fx2 – CAN it!* board is operating in normal mode.
Action:
Set the board to the FLASH programming mode by setting SW4 switch S1 to ON.

Cause:
The PRM file selected in [Device Setup] may be incorrect.
Action:
Use the corresponding PRM file that matches the target device. For information about the PRM file, refer to **CHAPTER 9 FPL3 FLASH programming software.**

Cause:
The setting of "Supply oscillator" in [Device Setup] may be incorrect.
Action:
Make a correct setting according to the specifications of the target device.

12. IAR configuration for On-Board debugging

12.1 Setup COM port for IAR C-SPY debugger

Before using the On-Board debug function of the *78K0/Fx2 – CAN it!* board together with the IAR C-SPY debugger it is necessary to set the corresponding USB serial COM port of the host computer. Just be sure that you have installed the USB driver first.

To set the COM port, please execute the registry script file “**FTDI_Set_Registry.reg**”. The script file can be found in the typical installation path of the IAR Embedded Workbench, i.e.:

“C:\Program Files\IAR System\Embedded Workbench 4.0\78k\config\nec”

This script file will set the COM port number in the following registry key:

[HKEY_LOCAL_MACHINE\SOFTWARE\NEC Tools\NECDEV\78K0.EX]

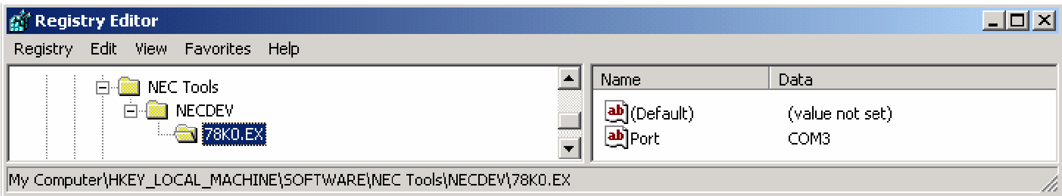


Figure 65: Registry entry for IAR C-SPY debugger COM port setting

As default the serial port COM3 is set in the script file. If the *78K0/Fx2 – CAN it!* board is used on a different USB serial COM port, please edit the script file accordingly. To do so, open the script file with your preferred editor i.e. “NOTEPAD” and make changes as described in the file. Then run the script file.

12.2 Configuration of USB serial COM port

To optimize the communication between the IAR C-SPY debugger and the *78K0/Fx2 – CAN it!* board the USB serial COM port must be configured. To do so, open the “Device Manager”, which can be found in properties of “My Computer”. Then locate the USB serial COM port and select “Properties”, as show in figure below:

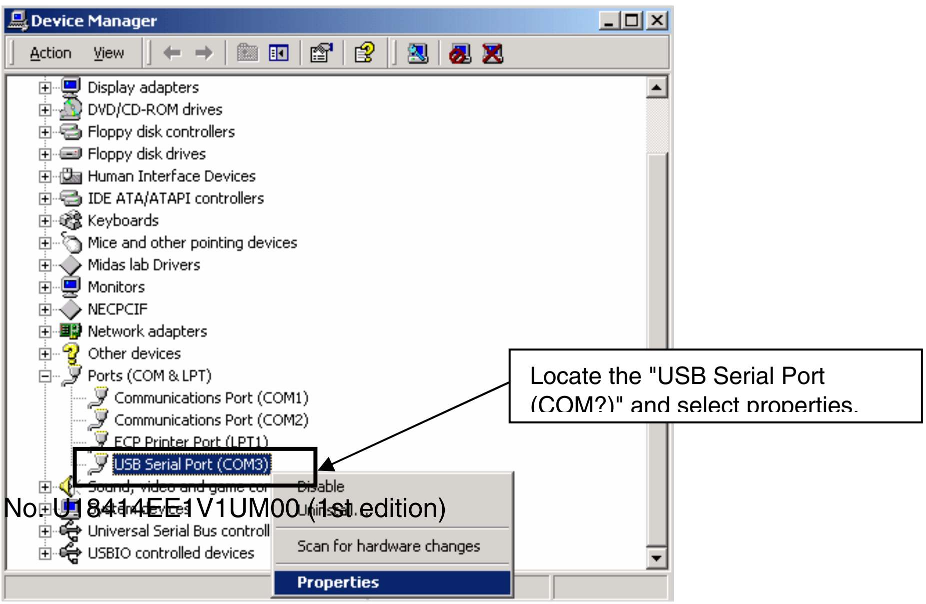


Figure 66: Configuration of COM port for IAR C-SPY debugger 1/4

The following USB Serial Port Properties Window appears:

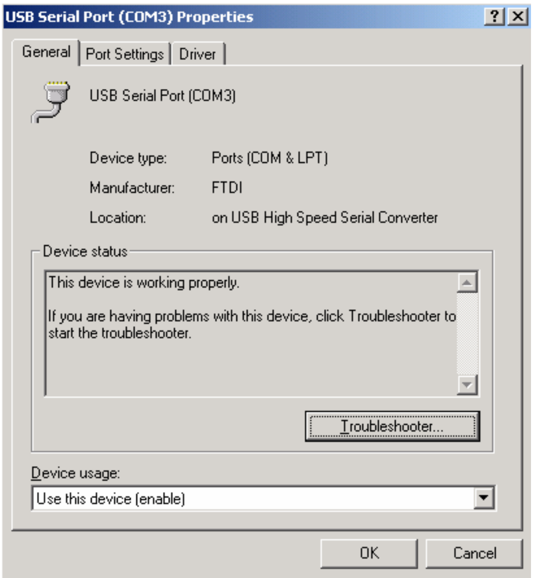


Figure 67: Configuration of COM port for IAR C-SPY debugger 2/4

Then select the “Port Settings” tab of the USB Serial Port Properties Window and press the “Advanced” button:

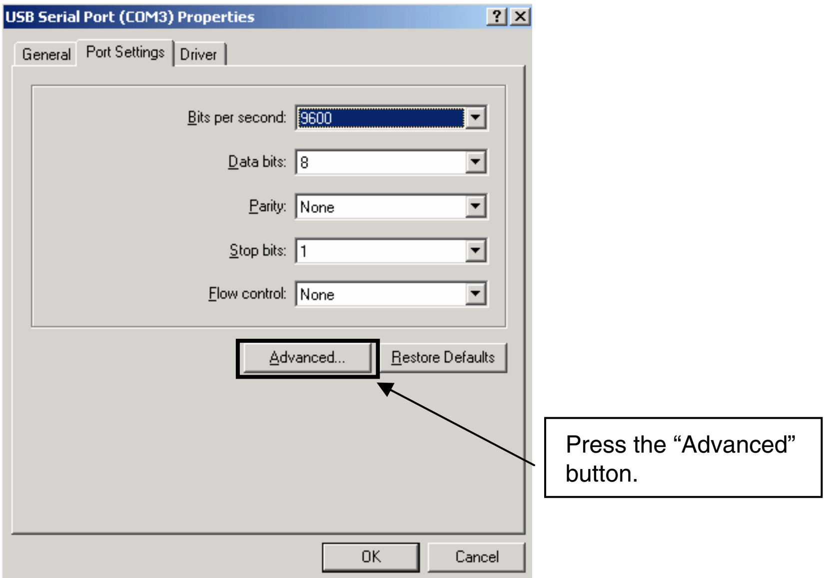


Figure 68: Configuration of COM port for IAR C-SPY debugger 3/4

Within the “Advanced Settings” menu change the “Latency Timer (msec)” to a value between 2 and 4. For faster host computers set the “Latency Timer” to value 1.

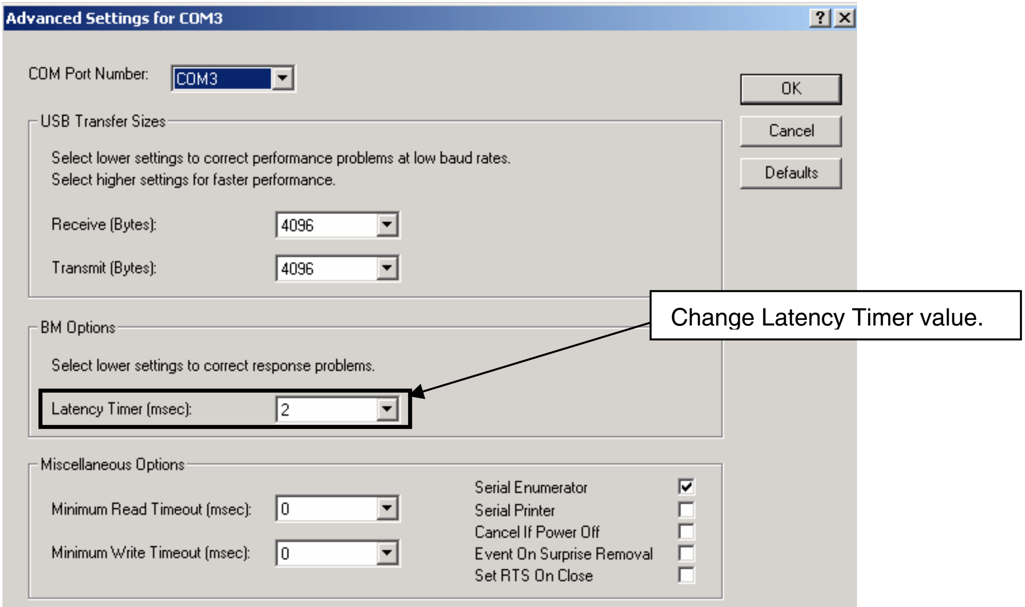


Figure 69: Configuration of COM port for IAR C-SPY debugger 4/4

Press the OK button to save the configuration and close the properties menu. The configuration of the USB serial COM port is finished now and the IAR C-SPY debugger can now be used in combination with the On-Board debug function of the *78K0/Fx2 – CAN it!* board.

13. IAR sample session

When everything is set up correctly the IAR Embedded Workbench can be started. To do so, start the Embedded Workbench from Windows "Start" menu > "Programs" > folder "IAR Systems" > "IAR Embedded Workbench Kickstart for 78K". The following screen appears:

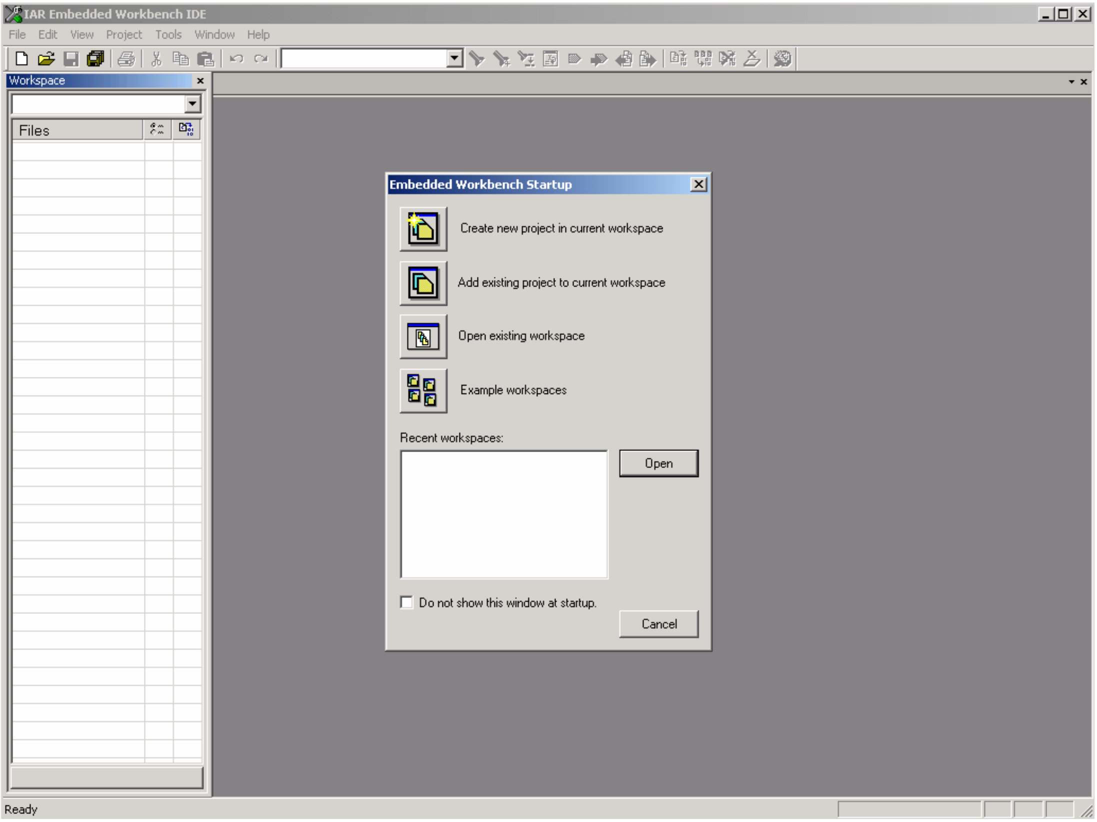


Figure 70: IAR Embedded Workbench

Now select the option "Open exiting workspace" from the "File" menu and locate the sample project. Open the file "78K0_CANit.eww". This is the workspace file that contains general information about the demonstration projects and settings.

After the demo workspace has been opened the projects contained in the workspace are displayed. Now click on the little “+” sign next to the “ADC_demo” project to show files that were part of the project. The screen should now look similar to this:

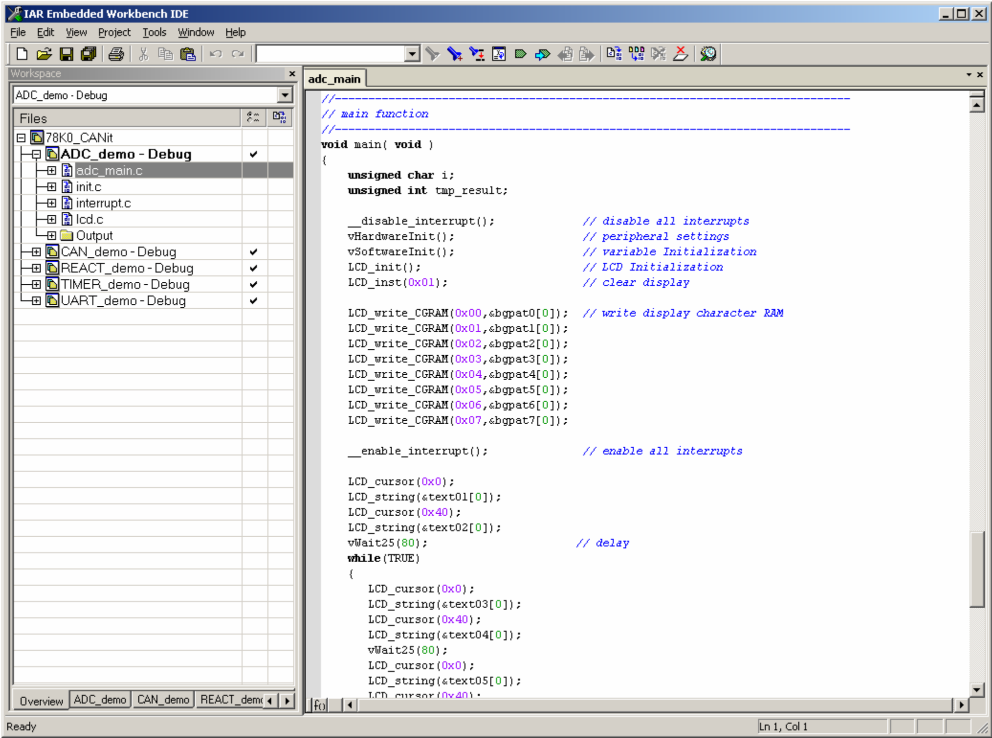


Figure 71: IAR project workspace

As a next step check some settings of the IAR Embedded Workbench that have to be made for correct operation and usage of the On-Board debug function of the *78K0/Fx2 – CAN it!* board. First highlight the upper folder called “ADC_demo – Debug” in the workspace window. Then select “Project” > “Options” from the pull-down menus. Next select the category “Debugger”. Make sure that the driver is set to “TK-78” in order to use the On-Board debug function of the *78K0/Fx2 – CAN it!* board. The device description file must be set to “io78F0893.ddf”.

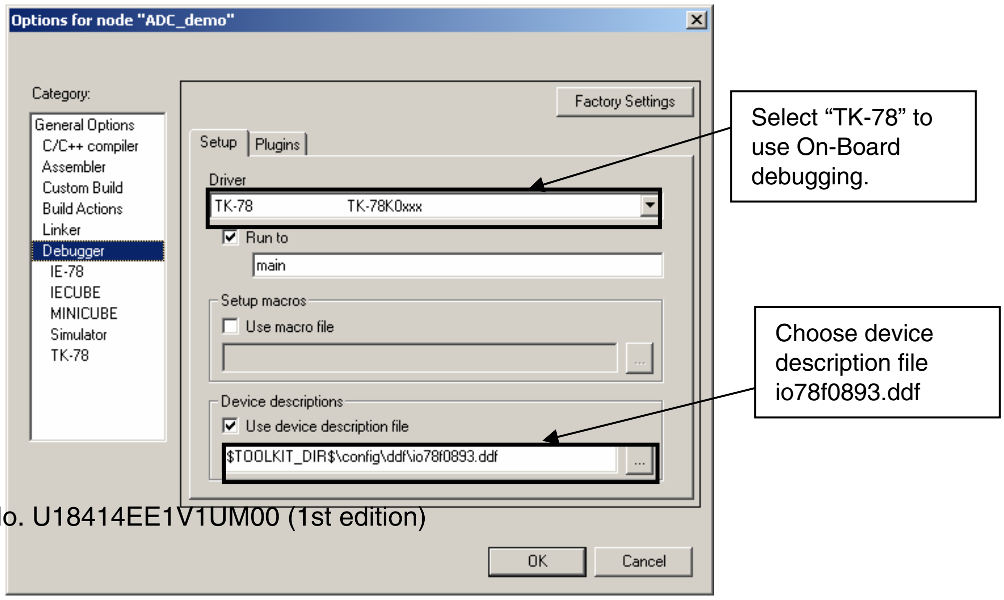


Figure 72: IAR debugger options

Next the correct linker settings of the demo project will be checked. This can be done in the "Linker" category as shown below. Select the "Config" tab and check that the linker command file "DF054780_V4.XCL" is selected. This file is used by the linker and contains information on where to place the different sections of code and data that may be used within the demo project:

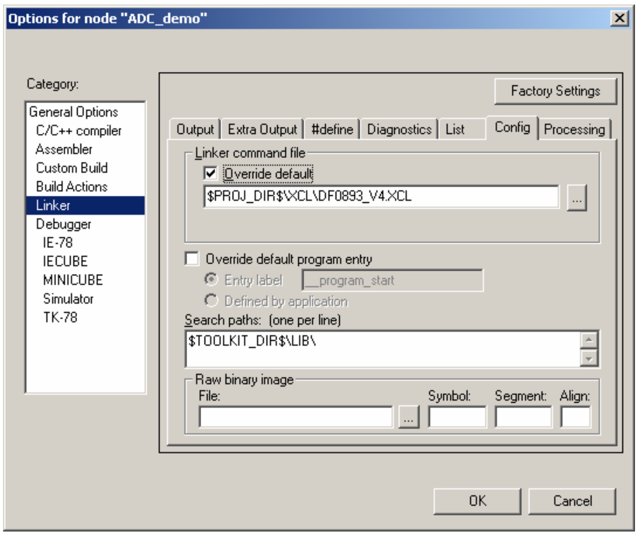


Figure 73: IAR Linker options

Now after everything has been setup correctly it's time to compile and link the demonstration project. Close the Options menu and select "Rebuild All" from the "Project" menu. If the project is compiled and linked without errors or warnings it can now be downloaded to the *78K0/Fx2 – CAN it!* board and debugged.

To start the IAR C-SPY debugger select the option "Debug" from the "Project" menu or press the ( ) "Debugger" button. In the next step the TK-78 Emulator has to be configured before downloading a new application. Press the OK button to enter the hardware setup menu. Set the configuration as show in the figure below and start the download by pressing the OK button.

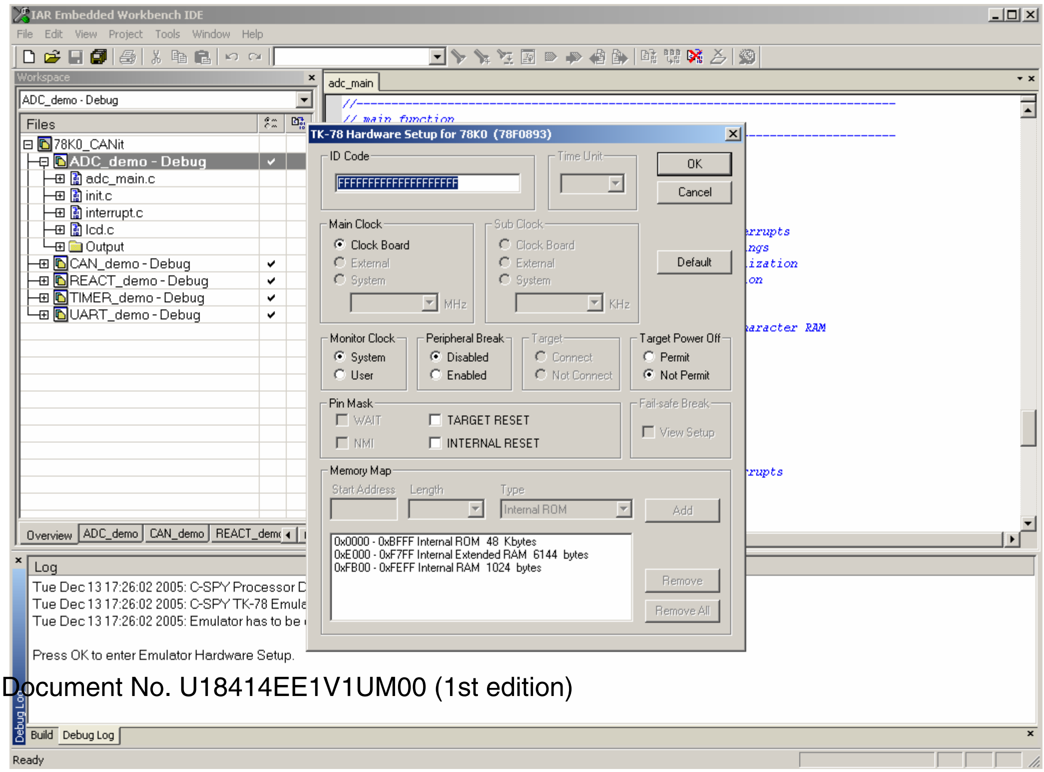


Figure 74: TK-78 hardware setup menu

Now the debugger is started and the demo project is downloaded to the *78K0/Fx2 – CAN it!* board. The progress of downloading is indicated by blue dots in the TK-78 Emulator window. Please note, downloading of greater executables can take some time.

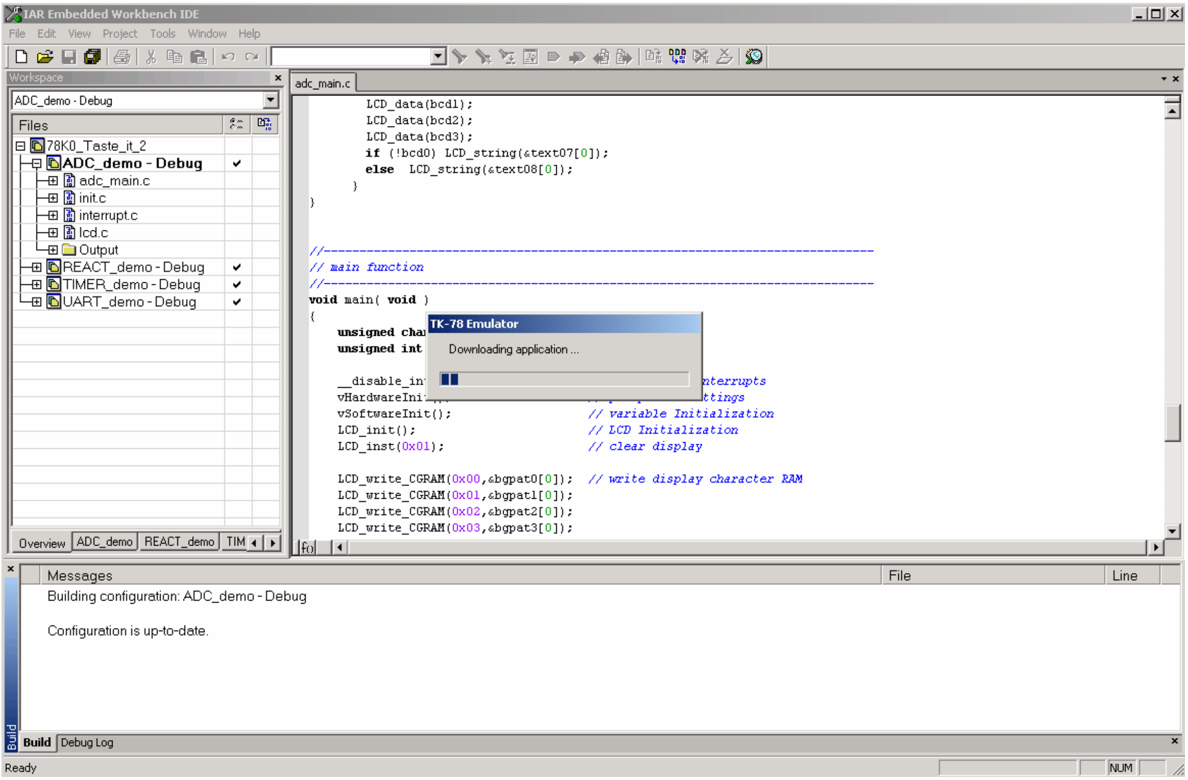


Figure 75: IAR project download

After the download was completed all debug features of IAR C-SPY debugger are available, i.e. Single Stepping (Step Over/-In/-Out), Go, Stop, Breakpoints, Register / Memory view etc.

To get more details on the debugger configuration and capabilities please refer to the “78K IAR Embedded Workbench IDE User Guide” of the IAR installation.

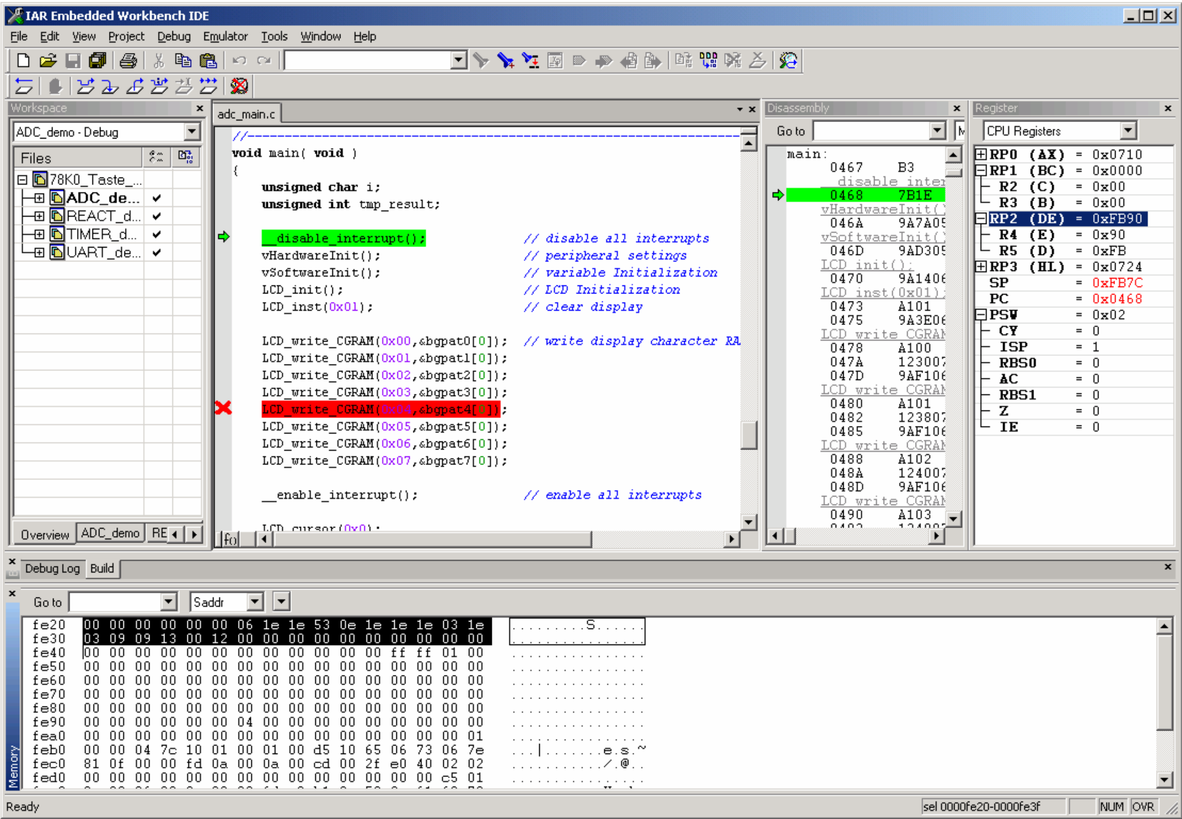


Figure 76: IAR C-SPY debugger

14. Sample programs

14.1 General Introduction

Each of the sample programs is located in a single directory, which will be called main-directory of the sample. This main directory of each sample contains the complete project inclusive all output files of the development tool. The workspace file “**78K0_CANit.eww**” is located on top of the sample program directories. All sample programs use the same directory structure:

AD Converter Demonstration	78K0/FF2 project and output files
Debug	debug output files for IAR C-SPY debugger
inc	C header files
Release	release output files, i.e. Intel HEX file
settings	configuration files, IAR Embedded Workbench
source	C source files
xcl	Linker control file
ADC_demo.dep	dependency information file, IAR Embedded Workbench
ADC_demo.ewd	project setting file, IAR C-SPY debugger
ADC_demo.ewp	project file, IAR Embedded Workbench
CAN Demonstration	78K0/FF2 project and output files
Reaction Time Measurement	78K0/FF2 project and output files
Timer Demonstration	78K0/FF2 project and output files
UART Demonstration	78K0/FF2 project and output files
78K0_CANit.eww	workspace file, IAR Embedded Workbench 78K0/78K0S

Table 21: Example directory structure

The main directory contains only the project files for the IAR Systems Embedded Workbench 78K0/78K0S. All source files are located in the directory `/source` and the `/inc` directory contains the header files. The `/xcl` directory contains the linker control file of the 78K0/FF2 device. Each sample project uses two targets. One target is the “Debug” (directory `/Debug`) that holds all information for debugging purpose and the other one the “Release” target (directory `/Release`) contains the programmable file, i.e. the Intel HEX file, for programming the 78K0/FF2 internal FLASH memory.

All output files of the development tools for the corresponding target are generated in the directories `/Debug` and `/Release`.

For details of using the IAR Embedded Workbench and the IAR C-SPY debugger please refer to the “78K IAR Embedded Workbench IDE User Guide”.

14.2 ADC demo

This sample program realizes a simple voltage meter. By using the integrated ADC, the voltage supplied to ADC on input channel 0, port P80 of 78K0/FF2 device, is measured. The input voltage is adjusted by potentiometer R14. The board shows the measured voltage by driving the LC display. The output format can be changed by pressing button SW2.

Used Internal Peripherals	Used External Parts
Timer00	Character LC display
Timer01	Potentiometer R14
Timer50	Button SW2
A/D converter	Button SW3
	LED's D1- D4

14.3 CAN demo

This sample program simulates a simple voltmeter as a CAN node. If the CAN node is used as the voltmeter, the voltage supplied to the integrated ADC (input channel 0, port P80 of 78K0/FF2 device) is measured and transmitted via CAN-bus in 100ms intervals. The input voltage is adjusted by potentiometer R14. The board shows the measured voltage by driving the LC display.

If the CAN node is used as display, the voltage value received via CAN-bus is displayed on the LCD. The configuration of the CAN node as transmitter or receiver can be done by buttons SW2 and SW3 during program start-up.

Used Internal Peripherals	Used External Parts
Timer00	Character LC display
Timer01	Potentiometer R14
Timer50	Button SW2
A/D converter	Button SW3
CAN controller	LED's D1- D4

14.4 ReacTime demo

This sample program demonstrates a reaction time measurement. After the button SW3 is pressed the application waits for a random time duration between 0.50 and 3.45 seconds. Then the measurement starts by incrementing a reaction counter every 50ms. The actual counter value is shown by a bar graph using the character LC display until the next keystroke of button SW3. When the button SW3 is pressed again, the measurement stops and the reaction time is shown on the LC display. Pressing button SW2 starts a new measuring cycle.

Used Internal Peripherals	Used External Parts
Timer00	Character LC display
Timer01	Button SW2
Timer50	Button SW3
	LED's D1- D4

14.5 Timer demo

This sample program demonstrates a real time clock. After the program is started the watch timer is initialized to generate an exact clock reference, based on the 32.768 kHz sub-clock. The actual time is displayed using the LCD module. By pressing button SW2 / SW3 the time can be adjusted.

Used Internal Peripherals	Used External Parts
Timer00	Character LC display
Timer01	Button SW2
WatchTimer	Button SW3
	LED's D1- D4

14.6 UART demo

This sample program realizes a temperature meter with serial communication channel. The sample program does a cyclic measurement of the input voltage of ANI2 analogue input, equal to port P82 of 78K0/FF2 device, connected to the temperature sensor KTY13-5. The temperature is calculated and the result is transferred via UART60 to a terminal program running on the host machine. The data transfer speed is set to 115200 bps per default. Additionally the temperature is displayed on the LCD. Before the temperature measurement is started the calibration of the temperature sensor has to be done, by adjusting the temperature offset via button SW2 and SW3.

Used Internal Peripherals	Used External Parts
Timer00	Character LC display
A/D converter	Temperature sensor KTY13-5
UART60	Button SW2
	Button SW3
	LED's D1- D4

Please note, by using the On-Board debug mode serial communication via UART60 is not possible because the USB/UART communication channel is used for debugging purpose.

15. USB interface cable (Mini-B type)

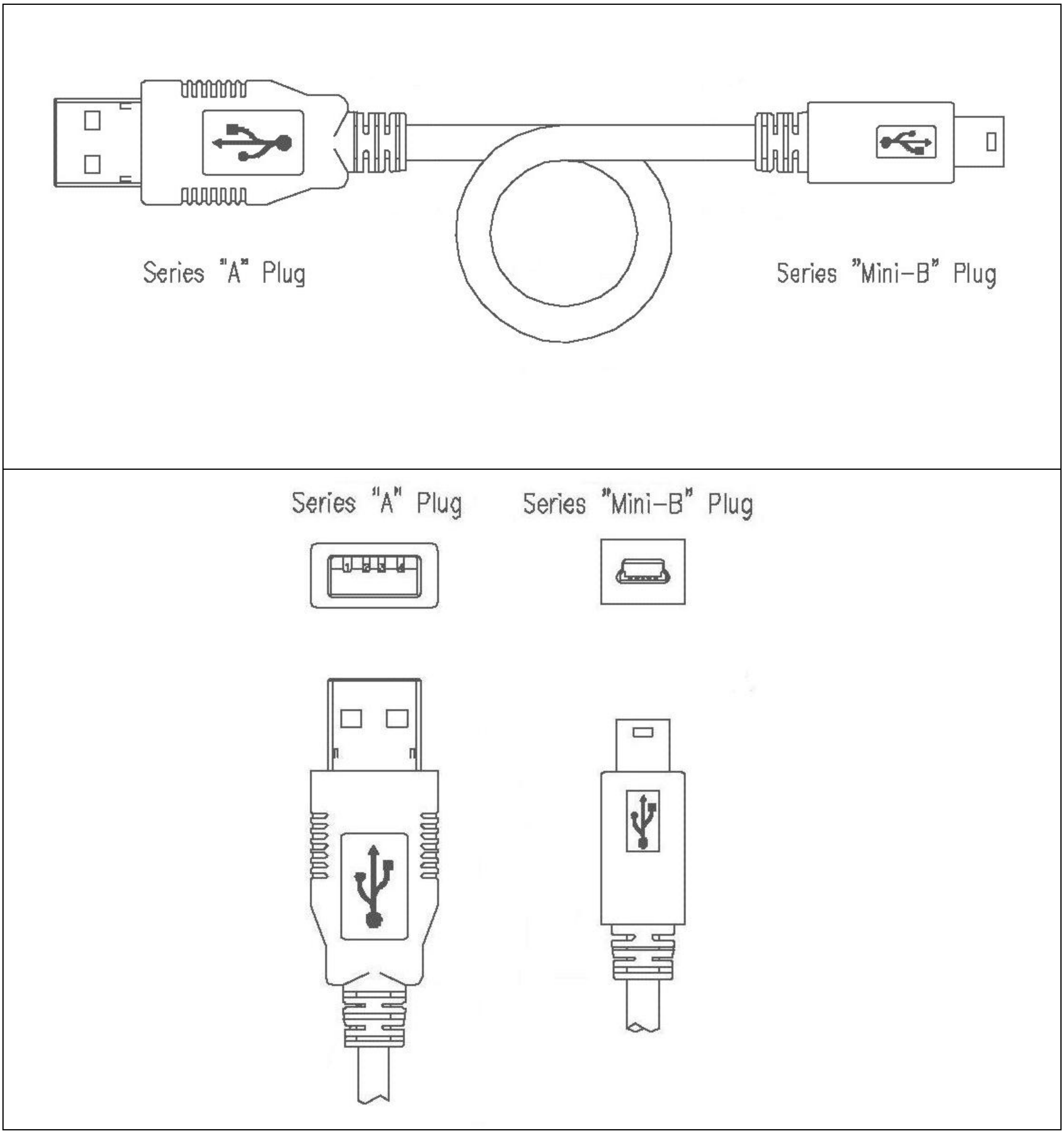
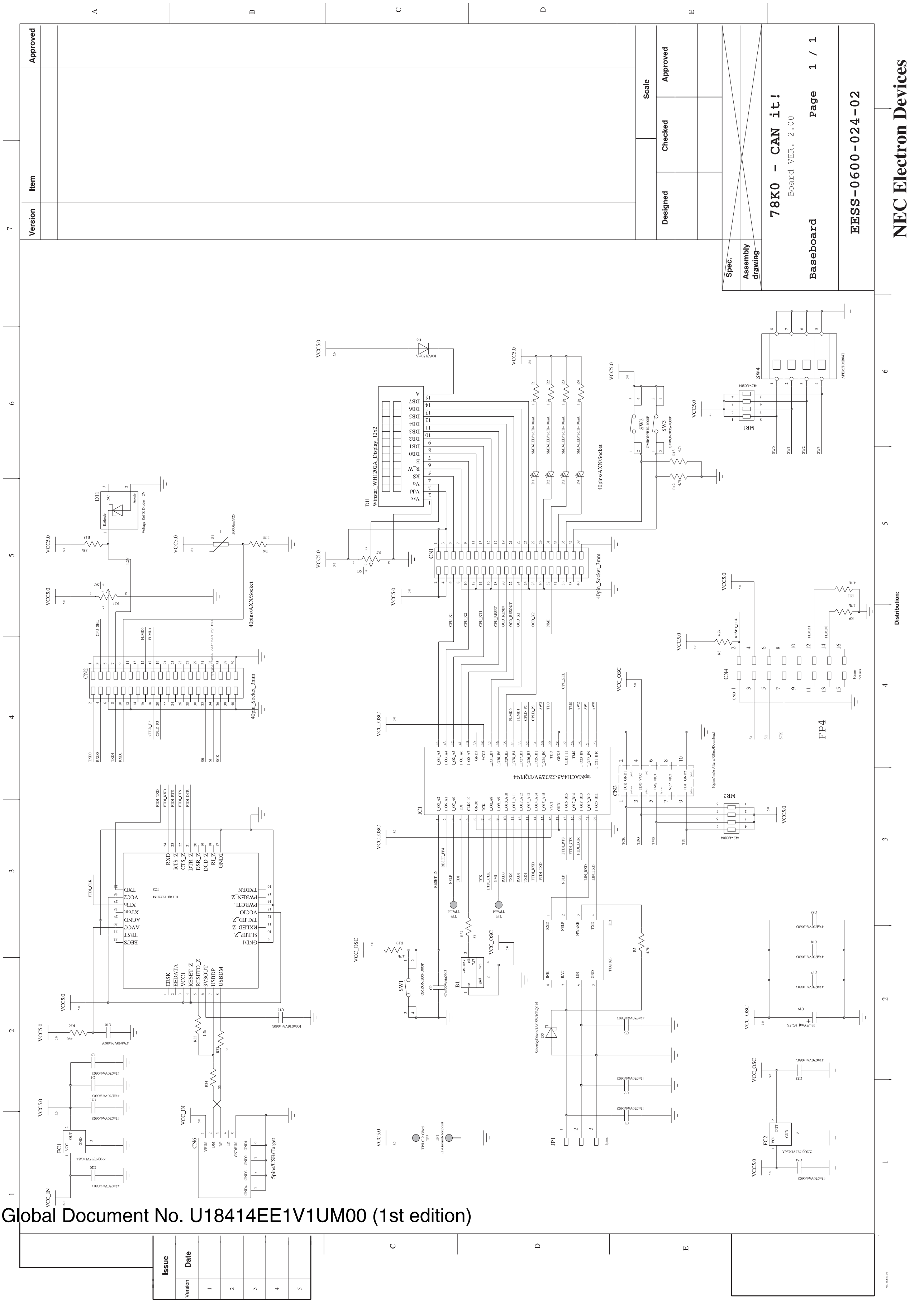


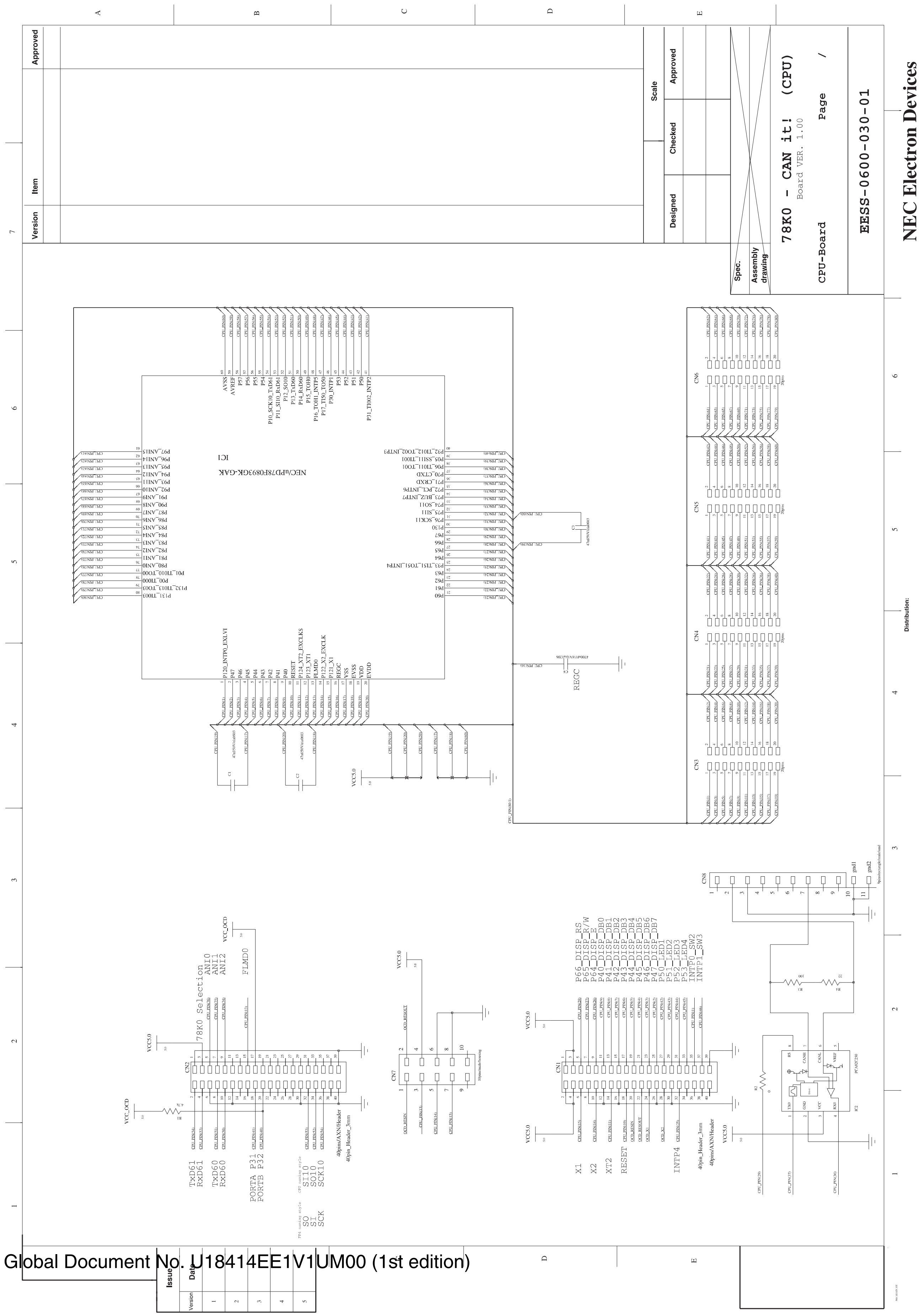
Figure 77: USB interface cable (Mini-B type)

16. Schematics



Global Document No. U18414EE1V1UM00 (1st edition)

Figure 78: ***78K0/Fx2 – CAN it!*** **baseboard schematics**



Global Document No. U18414EE1V1UM00 (1st edition)

Figure 79: *78K0/Fx2 – CAN it!* CPU module schematics

[MEMO]

Global Document No. U18414EE1V1UM00 (1st edition)